11-6-2017 2:00 PM

Efficiency and Accuracy Enhancement of Intrusion Detection System Using Feature Selection and Cross-layer Mechanism

Mahsa Bataghva
The University of Western Ontario

Supervisor
Dr. Xianbin Wang
The University of Western Ontario

Graduate Program in Electrical and Computer Engineering
A thesis submitted in partial fulfillment of the requirements for the degree in Master of Engineering Science

Abstract

The dramatic increase in the number of connected devices and the significant growth of the network traffic data have led to many security vulnerabilities and cyber-attacks. Hence, developing new methods to secure the network infrastructure and protect data from malicious and unauthorized access becomes a vital aspect of communication network design. Intrusion Detection Systems (IDSs), as common widely used security techniques, are critical to detect network attacks and unauthorized network access and thus minimize further cyber-attack damages. However, there are a number of weaknesses that need to be addressed to make reliable IDS for real-world applications. One of the fundamental challenges is the large number of redundant and non-relevant data. Feature selection emerges as a necessary step in efficient IDS design to overcome high dimensionality problem and enhance the performance of IDS through the reduction of its complexity and the acceleration of the detection process. Moreover, detection algorithm has significant impact on the performance of IDS. Machine learning techniques are widely used in such systems which is studied in details in this dissertation. One of the most destructive activities in wireless networks such as MANET is packet dropping. The existence of the intrusive attackers in the network is not the only cause of packet loss. In fact, packet drop can occur because of faulty network. Hence, in order detect the packet dropping caused by a malicious activity of an attacker, information from various layers of the protocol is needed to detect malicious packet loss effectively. To this end, a novel cross-layer design for malicious packet loss detection in MANET is proposed using features from physical layer, network layer and MAC layer to make a better detection decision. Trust-based mechanism is adopted in this design and a packet loss free routing algorithm is presented accordingly.

Acknowledgment

First and foremost, I am very grateful to my academic advisor, Professor Xianbin Wang, for accepting me into his group, for his support, patience, and encouragement throughout my graduate study. His technical advice was essential to the completion of this dissertation and has taught me innumerable lessons and insights on the workings of academic research in general. During my graduate studies, he contributed to a rewarding graduate school experience by giving me intellectual freedom in my work, engaging me in new ideas, and demanding a high quality of work in all my endeavors.

Additionally, I would like to thank my examination committee members, Dr. Primak, Dr. Samarabandu and Dr. Haque for their interest in my work and their invaluable guidance and support.

I would also like to thank my parents for providing me with continuous encouragement and unconditional love and support throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without their endless support. Their endless love, support and encouragement was in the end what made this dissertation possible.

I would also like to acknowledge the Department of Electrical and Computer Engineering at Western University. My graduate experience benefited greatly from the courses I took, the opportunities to serve as a teaching assistant and to learn from the high-quality seminars that the department organized.

Last, but not least, I would like to thank my colleagues, lab mates and friends.

Contents

Abstract ii

Acknowledgment iii

List of Figures vii

List of Tables viii

List of Abbreviations ix

1 Introduction 1
- 1.1 Background and Problem Statement . . . 1
- 1.2 Research Motivation and Objectives . . . 4
- 1.3 Dissertation Contribution . . . 4
- 1.4 Dissertation Organization . . . 8

List of Abbreviations 1

2 Introduction to Network Security Challenges and Solutions 10
- 2.1 Security Techniques in Wireless Communication Networks . . . 11
  - 2.1.1 Security Prevention Techniques . . . 13
    - Cryptography . . . 13
    - Authentication . . . 14
  - 2.1.2 Intrusion Detection Systems . . . 14
    - Classification of Intrusion Detection Systems Based on the Detection Approach . . . 17
    - Classification of Intrusion Detection Systems Based on the Place of Deployment . . . 22
  - 2.1.3 Security Mitigation Techniques . . . 23
- 2.2 Related Works and Suggested Studies . . . 24

3 Feature Selection in IDS 25
- 3.1 Introduction . . . 25
- 3.2 Feature Selection Schemes . . . 30
- 3.3 Proposed Feature Selection Model to Increase Performance Efficiency . . . 35
  - 3.3.1 Proposed Feature Selection Method . . . 35

Correlation-based Feature Selection Technique to Remove Redundant Features . . . 35
Symmetrical Uncertainty Technique to Remove Irrelevant Features . . . 38
3.3.2 Proposed Feature Selection Algorithm . . . 39
3.4 Simulation Results and Performance Evaluation . . . 41
3.4.1 NSL-KDD Dataset . . . 43
3.4.2 Evaluation Metrics . . . 45
3.4.3 Experimental Results and Analysis . . . 47
3.5 Summary . . . 52

4 Intrusion Detection Using SVM and RVM Classification Algorithms 53
4.1 Introduction . . . 53
4.2 Support Vector Machine Technique for IDS . . . 56
4.3 Kernel Functions and Their Affect on SVM . . . 63
4.4 Relevance Vector Machine Technique for IDS . . . 66
4.4.1 Training Phase of the Learning Process . . . 68
4.4.2 Classification Phase of the Testing Process . . . 70
4.5 Performance Comparison Between Support Vector Machine Technique and Relevance Vector Machine Technique . . . 70
4.5.1 Performance Comparison in Terms of Sparsity . . . 74
4.5.2 Performance Comparison in Terms of Generalization . . . 75
4.5.3 Performance Comparison in Terms of Classification . . . 75
4.5.4 Performance Comparison in Terms of Complexity . . . 76
4.6 Summary . . . 77

5 Cross-layer Trust-based IDS for Malicious Packet Loss 78
5.1 Introduction . . . 78
5.1.1 Cross-layer Intrusion Detection Technique . . . 83
Multiple Data Collection Multiple Data Analysis . . . 90
Multiple Data Collection Single Data Analysis . . . 90
5.1.2 Trust-based Compromised Node Detection . . . 91
5.2 Attack Types and Categories . . . 92
5.2.1 Active Attacks Versus Passive Attacks . . . 92
5.2.2 Internal Attacks Versus External Attacks . . . 93
5.2.3 Layer-specific Attacks . . . 94
5.3 Proposed Cross-layer Trust-based Malicious Packet Loss Detection Architecture 95
5.3.1 Data Collection and Feature Extraction . . . 95
Cross-layer Features . . . 96
5.3.2 Intrusion Detection Based on Trust Model . . . 98
Trust Model Generation . . . 99
Malicious Packet Loss Detection . . . 106
5.3.3 Response Module . . . 107
5.4 Proposed Secure Path Selection . . . 107
5.4.1 Observer Node Selection . . . 109
5.5 Performance Evaluation . . . 109

5.5.1 Validation of Trust Value Evaluation . . . 110
5.5.2 Performance Metrics . . . 111
5.6 Simulation Results . . . 112
5.6.1 Assumption . . . 112
5.6.2 MANET Routing Protocol . . . 113
5.6.3 Simulation Environment . . . 116
Detection Performance . . . 116
Secure Path Performance . . . 118
5.7 Summary . . . 119

6 Conclusion and Future Works **121**
6.1 Conclusion . . . 121
6.2 Future Works . . . 124

Bibliography **126**

List of Appendices **134**

Curriculum Vitae **152**

List of Figures

2.1 Security Provisioning for Wireless Networks 12

3.1 Intrusion detection system (IDS) model 36
3.2 Testing phase of the intrusion detection system in WEKA 39
3.3 Training times for different feature selection techniques 49
3.4 Performance comparison among different feature selection techniques 50

4.1 Support Vector Machine Classifier Demonstration 58
4.2 Relevance Vector Machine Classifier Demonstration 68
4.3 Comparison of the Sparsity Ability of SVM and RVM 75
4.4 Comparison of the Generalization Ability of SVM and RVM 76

5.1 MANET Topology 80
5.2 Proposed Cross-layer IDS Architecture 96
5.3 Secure Path Selection Based on Trust-based Cross-layer AODV 110
5.4 Performance of Proposed Trust-based Cross-layer IDS in terms of False Positive Rate 117
5.5 Comparison of Packet Delivery Ratio of Proposed Trust-based Cross-layer IDS with Other Packet Loss Detection Technique 118

List of Tables

3.1 Confusion Matrix considering abnormal classes as a target 43
3.2 Confusion Matrix considering normal classes as a target 43
3.3 The performance of feature selection techniques with different classifier in terms of detection rate 48
3.4 The performance of feature selection techniques with different classifier in terms of false positive rate 48
3.5 Classification results using only 4 features of NSL-KDD dataset 51
3.6 Classification results using all 42 features of NSL-KDD dataset 51

4.1 Effect of Gamma Using RBF kernel function in SVM on NSL-KDD dataset . . 65
4.2 Effect of Kernel Functions on SVM Without Feature Selection on NSL-KDD Dataset 67
4.3 Classification Comparison between SVM and RVM 76

5.1 DoS attacks in different layers of the protocol 93
5.2 Parameters in different layers of the protocol 94
5.3 Trust Table 110
5.4 Network Configuration for Performance Analysis 113

List of Abbreviations

Acc	Accuracy
ACK	Acknowledgement
AODV	Ad-hoc On Demand Distance Vector
CFS	Correlation-based Feature Selection
CTS	Clear To Send
DoS	Denial of Service
DR	Detection Rate
DSR	Distance Source Routing
FAR	False Alarm Rate
FN	False Negative
FPR	False Positive Rate
FP	False Positive
GR	Gain Ratio
IDS	Intrusion Detection System
IG	Information Gain
MAC	Media Access Control
MANET	Mobile Ad-Hoc NETwork
MAP	Maximum Posteriori
MCSA	Multiple data Collection Single data Analysis
MCMA	Multiple data Collection Multiple data Analysis
MIFS	Mutual Information Feature Selection
MMH	Maximal Margin Hyperplane
ML	Maximum Likelihood
MR	Miss Rate
NMIFS	Normalized Mutual Information Feature Selection

OSI	Open System Interconnection
PDR	Packet Delivery Ratio
QoS	Quality of Service
QP	Quadratic Programming
RBF	Radial Basis Function
ROC	Receiver Operating Characteristic
RMSE	Root Mean Square Error
RREQ	Route Request
RERR	Route Error
RREP	Route Reply
RSS	Recieved Signal Strength
RTS	Request To Send
RVM	Relevance Vector Machine
RV	Relevance Vector
SBL	Sparse Bayesian Learning
SU	Symmetrical Uncertainty
SVM	Support Vector Machine
SV	Support Vector
TN	True Negative
TP	True Positive

Chapter 1

Introduction

1.1 Background and Problem Statement

The rapid growth of the computer network activities and the significant growth in the number of computers and electronic devices have increased the rate of network attacks. Due to open environment in wireless communication, network security becomes even more crucial for wireless networks. The attacks have become more sophisticated with more severe effect and are much harder to be identified. Moreover, the growth in Internet application has made it extremely difficult to process and analyze the Internet network traffic flow to find abnormality or attacks in the network. In addition, since cyber-attacks that attempts to damage or destroy computer systems and networks are increasing every day, network security becomes even more crucial. Cyber-attacks are de- fined as any set of actions that cause to alter, disrupt, deceive, degrade, or destroy adversary computer systems and networks. Security methods for wireless networks can be categorized into three types, namely, prevention, detection and mitigation techniques. Any attempt that tries to secure the wireless network to prevent attacks from happening is called

prevention techniques. Authentication, cryptographic techniques, key exchange are some examples of prevention techniques. The conventional cryptography techniques are very effective in preventing the network from external attackers. However, they cannot detect an internal node which provides false routing information or a node that does not cooperate with other nodes. Moreover, if an attacker could bypass the prevention step, meaning that prevention techniques are not sufficient to provide security to the system against the attackers, intrusion detection systems as the second line of defense comes into play. Whenever the network is aware of the presence of an attacker, it is of a great importance to identify the attacker accurately and quickly. In this phase, the network tries to find the compromised nodes quickly and accurately to prevent further harmful damages. Intrusion detection system is one of the very important mechanisms being applied for this phase. However, current IDS suffers from low detection accuracy and high false detection rate. Since the security level of the network, after prevention techniques, relies on the performance of IDS and since IDS with higher malicious behavior detection rate can indirectly provide more secrecy to the wireless networks, it is vital to enhance IDS performance. Being motivated to improve the performance of intrusion detection systems, in this dissertation, the parameters that affect IDS performance are investigated in details. The performance of IDS is affected by the features from the network traffic data that are used to discriminate the network behavior into normal and abnormal as well as the classification algorithm that is used to separate the normal network behavior from abnormal behavior. Almost no intrusion detection can distinguish intrusive network connection from normal ones directly from captured original packets. Feature extraction and feature selection are necessarily required to obtain useful information from raw data. Moreover, extraction of proper features can enhance the detection mechanism significantly. On the other hand, using

machine learning techniques such as classification algorithms are commonly used in IDSs to distinguish different classes of network behavior and categorize them accordingly. Since the essence of both feature selection techniques and classification algorithms in IDS as two major components that can affect overall IDS performance significantly is obvious, the effect of both above-mentioned contributing factors are studied in details in this dissertation. On the other hand, since the performance of single-layer IDS is not sufficient, using only one layer features in IDS or in other words, having IDS reside in a single layer of the OSI protocol does not provide desirable detection results, Moreover, having IDS reside in only one layer cause the detection system be sensitive to the attacks associated with that particular layer and neglect the attackers that aims other layers than where the IDS resides. Therefore, a cross- layer IDS as an approach to utilize features from multiple layer of the protocol instead of a traditional single layer attributes can be an effective solution and is studied in this dissertation. Being motivated to observe the application of IDS is real-word network malicious behavior detection, the scenario of malicious packet loss detection in mobile Adhoc networks is presented. However, instead of single layer IDS, a cross-layer mechanism is adopted to use multiple layers of the protocol to detect malicious packet loss more effectively. Moreover, to consider non-malicious causes of packet loss, a trust-based model is used to help distinguishing between malicious and non-malicious causes of packet loss. Trust-based security mechanisms is another well-known detection technique that have been used in many studies and are one of the popular techniques to enhance the security of the routing protocols. Moreover, after identifying the malicious nodes that drops the network packets intentionally to cause disruption in the network, the secure routing path from source to destination is established using trust values of the nodes and the proposed cross-layer IDS.

1.2 Research Motivation and Objectives

As mentioned earlier, there are several limitations and shortcomings with current intrusion detection systems. In this dissertation the promising solutions to overcome the problem of high dimensionality as well as low accuracy and the problems associated with the single-layer IDS are proposed. The main research objectives of this dissertation can be categorized in three folds as follows:

1. To enhance the overall intrusion detection system performance in terms of reducing the required time to train and test the classification model, classification accuracy and effectiveness as well as complexity minimization by proposed feature selection method;

2. To analyze the effect of kernel functions and their related parameters on two well-known classification algorithms, namely, support vector machine and relevance vector machine to enhance the classification performance and compare their performance of in terms of classification effectiveness and efficiency as well as applicability in terms of sparsity and generalization capability;

3. To use multiple layers of the OSI protocol and use their related features in identifying and detecting attacks more efficiently. Combination of attributes from proper layers can help in achieving a significant enhancement the detection performance;

1.3 Dissertation Contribution

The need for feature selection techniques is vital since feature selection identifies the most informative features and derive a subset of network traffic attributes that is free of irrelevancy

and redundancy. The selection of features will enhance the efficiency of intrusion detection. Moreover, feature selection reduces the complexity and training time since only a subset of all features with less number of attributes from a whole dataset is used for detection of malicious behavior in the network. The main contributions in the third chapter are:

1. A new feature selection techniques based on the correlation among feature and the symmetrical uncertainty is proposed;

2. The performance of the proposed feature selection technique is compared to other feature selection techniques;

3. The performance of various well-known classification algorithms using subset of features selected based on proposed feature selection algorithm are studied and compared;

4. The results illustrate that the proposed attribute selection technique with less number of features not only results in less training time and less computation complexity but it improves the performance in terms of accuracy and detection rate;

It is notable that with more number of features the accuracy increases but with the cost of computational complexity and training time. However, if the number of contributing features in defining the secrecy of the network traffic behavior is selected properly, one can reduce the training time significantly to provide close to real-time training and decrease the complexity remarkably. It has been proved that the reduction in number of features can degrade the performance since less number of features provide less amount of information about the network traffic behavior. On the other hand, if the right set of features are selected properly, not only the intrusion detection becomes more efficient in terms of time consumption and com-

plexity but also the detection accuracy can remain similar to using all the attributes. In other words, feature selection maximizes the efficiency while it does not degrade the effectiveness (e.g. detection accuracy) of the malicious network behavior detection, since there is a trade-off between detection accuracy and training efficiency.

On the other hand, since classification algorithm is the core element of any intrusion detection, the selection of right classifier plays an important role in the detection accuracy and the overall performance of the intrusion detection algorithm. In the fourth chapter, we focused on investigation of the two well-known classification algorithms: Support Vector Machine (SVM) and Relevance Vector Machine (RVM). First, we tested their performance on the well-known NSL-KDD data set in which the classifier is responsible to categorize the normal and anomaly network traffic instances accurately. In addition, since the performance of above-mentioned classifiers highly relies on the selection of their parameters, the effect of different Kernel Methods and their parameters on SVM classification performance on NSL-KDD data set for intrusion detection purposes is investigated. Moreover, a comprehensive comparison between SVM and RVM classifier in terms of detection performance, sparsity ability as well as generalization and effectiveness is presented. The contribution in the fourth chapter are:

1. The effect of different type of kernel functions are studied on SVM and their performance are compared in terms of accuracy, precision, F-measure and the area under ROC curve;

2. RVM and SVM are compared in terms of their sparsity property and the classifiers are tested with different portion of the training set and the number of required support vectors and relevant vectors are derived to compare the sparsity ability accordingly;

3. RVM and SVM are compared in terms of generalization property and classifiers are

tested with difference portion of the testing set and the number of required support vectors and relevant vectors are derived to compare the generalization ability;

4. The required training time and complexity as well as classification performance in terms of detection accuracy are studied for both SVM and RVM and compared accordingly;

In the fifth chapter, proposed cross-layer trust-based IDS for routing in MANET is studied in details. In this Chapter, the objective is, first, to identify malicious packet dropping accurately with less false positive rate and second, to obtain a secure and reliable path to enhance the efficiency and performance of MANETs free of both malicious and legitimate packet loss attacks. Effective packet loss detection and providing secure and reliable routing path in the network is proposed in this chapter using cross-layer trust-based detection system which is based on the AODV protocol. Comparing to single layer IDS, it has significantly higher detection rate and accuracy since it uses attributes and features from various layers of the protocol to make a decision about the performance of a node in the network. Different causes of packet loss to reduce false positive rate in better identifying malicious packet loss in the network is considered. The four contributions of this chapter are:

1. Extracted of informative feature from multi layers to define network traffic behavior and be used in cross-layer intrusion detection system design;

2. Built a trust model based on legitimate causes of packet loss using cross-layer features;

3. Proposed a new cross-layer trust-based malicious packet loss detection algorithm;

4. Proposed a new secure routing path algorithm free of packet loss attack;

There are two major contributions in our work comparing to similar studies available in the literature. First, almost all of the constraints and limitations of the MANETs are taken into consideration which is not the case in many research studies. Second, the performance of the intrusion detection technique has considerably improved with proposed approach.

1.4 Dissertation Organization

The thesis is organized as follows:

In chapter II, a detailed background on security techniques for wireless communication is provided. Chapter II an overview on prevention techniques, followed by a full description of the detection techniques and their limitations and constraints. Different intrusion detection techniques are studied in details and the advantages and drawbacks of each approach is presented. Finally mitigation techniques are studied.

In chapter III, first, a literature review on feature selection techniques is provided. Second, the proposed feature selection technique to reduce redundancy and irrelevant features to decrease the complexity and training time is studied followed by feature selection algorithm. Next the simulation results on NSL-KDD data set that us very commonly used dataset for intrusion detection studies is presented with results and analysis discussion.

In chapter IV, two well-known classification algorithms, Support Vector Machine(SVM) and Relevance Vector Machine(RVM) and their mathematical view is studied in details. The effect of different parameters as well as kernel function as a very important factor on the classification performance is investigated and two classification algorithms are tested on NSL-KDD dataset and compared in terms sparsity, generalization, classification and decision speed.

Moreover, in chapter V, first, the notion of cross-layer IDS is explained. Second, different types of attacks against various layers of the node protocol security are discussed is studied precisely since without knowledge of attacks, neither can security measures be devised to protect wireless networks, nor can models be developed to detect intrusions. Therefore, more information on attacks and security measures to counter those attacks are provided. . Thirdly, cross-layer intrusion detection system is studied and a new cross-layer trust-based detection techniques to detect malicious packet loss attacks more accurately is proposed. The idea behind cross-layer IDS is to use attributes in multiple layer of the protocol to detect an adversary attack in one layer, here packet drop detection in network layer. Moreover, the proposed cross-layer IDS model and different components of the proposed model is studied. In addition a new secure and reliable routing path selection free of the risk of packet loss attack is introduced. Simulation results and performance analysis is provided.

Finally, the contributions from Chapters II to V are summarized and a brief conclusion and recommendations for future work are presented in chapter VI.

Chapter 2

Introduction to Network Security Challenges and Solutions

Exponential growth in the number of users from various heterogeneous devices has caused a significant increase in the number of attacks. One of the essential threats to the security of communication network is attacks by malicious individuals from inside and outside of the system to disrupt the available services, or reveal their confidential information. Therefore, security systems are required to be enhanced to be capable of detection of vast variety and sophisticated attacks.

The security mechanism is a procedure of prevention, identification and recovering of the network from security attacks. Security is basically a trade-off between the mitigation of breaches against the costs of security and both of which are difficult to measure. The main goal of security systems is to enhance the security of data and exchange information [2]. In any security system, if the first line of defense, prevention technique, does not help to secure the network from attackers, the second line of defense is required. The goal in any intrusion de-

tection system is to maximizing the probability of malicious nodes detected as attackers (True Positive Rate) while minimizing the probability of normal nodes being detected as attackers (False Positive Rate) [3]. In general, security solutions can be classified as cryptography methods, intrusion detection systems, and trust-based or reputation based solutions [4]. IDS is a method for attack detection by continuously monitoring and analyzing network functions [5].

2.1 Security Techniques in Wireless Communication Networks

The increase in a number of the computers and electronic devices as well as a number of attackers that threaten wireless networks are increasing rapidly. Moreover, attackers are becoming more complex and difficult to detect every day. Hence, a security mechanism for wireless networks is required to be adopted that can protect the wireless networks from intruders and attackers. The fact is that all of the security services in wireless multi hop networks have a similar objective which is to protect the network and transmitted information from attacks. All security techniques aim to ensure confidentiality, integrity, and availability of the net- work and they are the foundation of every security system [4].

Confidentiality Ensures the information would never reveal to unauthorized parties and devices. Confidentiality is also called as secrecy which is used to make the information unapproachable to unlawful users. Cryptographic techniques are used to ensure the confidentiality of the information transmission in the network;

Integrity Ensures that packets are transmitted without any modification in the content. In other words, the received data should be correct and valid and without any alteration in transit by an attacker. It verifies the content of the communicated data is ensured to be free from any

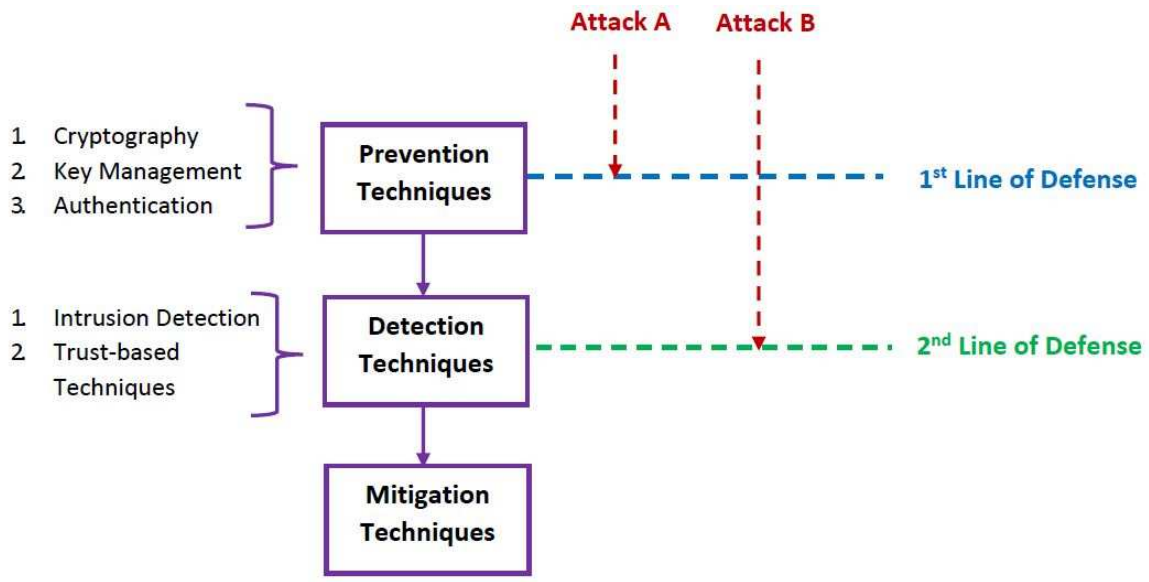

Figure 2.1: Security Provisioning for Wireless Networks

type of alteration between the end points. Information must not be corrupted, degraded, or modified. Simply, data integrity refers to the consistency and accuracy of data to ensure that unauthorized parties are prevented from modifying data;

Availability Ensures the sustainability of the network functionality without any interruption due to security threats to the authorized parties. It ensures if information can reach the destination via communication channel. Intrusion detection techniques are used to provide availability to the network; Intrusions are any set of actions that attempt to compromise the integrity, confidentiality, and availability of the network resources [6]. There are typically three levels of security mechanisms for wireless networks, namely: prevention techniques, detection techniques and mitigation techniques.

2.1.1 Security Prevention Techniques

Any defense strategies that are taken to secure the communication network before attacks are happening are called security prevention techniques. Any attempt that tries to secure the wireless network to prevent attacks from happening is called prevention techniques. Although many studies proposed various techniques and schemes for security prevention techniques, they cannot fully secure the networks. Authentication, cryptographic techniques and key exchange are some examples of prevention techniques. All the cryptographic techniques are within this category. Authentication, encryption, hashing and digital signature fall in this category. The conventional cryptography techniques are very effective in preventing the network from external attackers; however, they cannot detect an internal node which provides false routing information or a node that does not cooperate with other nodes.

Cryptography

Cryptography is the art of securing the message and is a technique in which data is stored and transmitted in a particular form so that only intended user can read and process it. Cryptography is concerned with developing different types of techniques that prevent reading private messages by the attackers. Any cryptographic techniques consist of encryption and decryption which are the process of converting original message into a form unreadable by unauthorized individuals and decoding the converted form of message to its original form to be readable to the authorized party, respectively. Key exchange is the most common techniques in cryptography. In this approach, a secret key is shared between only the transmitter and the receiver to encrypt and decrypt the data in a way that no other node in the network have the informa-

tion about the password or the key. In this way, even if the message is captured by any other node, it is useless since they cannot read the content of the message because it is encrypted. The two communicants, here the transmitter and the receiver, in secret key system require the prior communication of key, using a secure channel. However, it is very difficult to achieve in practice.

Authentication

Authentication is the process that ensures and confirms a user's identity. Authentication is the process which allows a sender and receiver of information to validate each other. If the sender and receiver of information cannot properly authenticate each other, there is no trust in the activities or information provided by either parties.

2.1.2 Intrusion Detection Systems

In the detection phase, however, the network is aware of the attack that is present. If an attacker manages to pass the measures taken by the prevention techniques, it means that there is a failure in the network that needs to be defeated. This is when detection strategies come into play to identify the nodes that are being compromised. Detection-based approaches have attracted lots of attentions in the last few years since prevention techniques for security vulnerability such as authentication and key-exchange methods are not capable of identifying and defending the insider attacks effectively. Therefore, the need for misbehavior detection plays a vital role in wireless networks. Intrusion detections have gained lots of attention among researchers due to their significance ability and performance. Whenever the network is aware of the presence of an attacker, the network tries to find the compromised nodes quickly and accurately to prevent

more harmful damages.

Intrusion detection system is one of the very important mechanisms being applied for this phase. Trust- based security mechanisms is another well-known detection technique that have been used in many studies and are one of the popular techniques to enhance the security of the routing protocols. Intrusion Detection was first introduced in [7]. IDS is a collection of tools, methods, and resources available to help to identify, assessing and reporting any intrusions in the network [8]. In [9], the authors defined IDS as a security mechanism that collects information from the net- work and by analyzing the gathered data to find abnormal behavior. Intrusion detection is defined as deployed systems that try to identify any set of actions that compromise the avail- ability, integrity, and confidentiality of the network resources [10]. In [11], the authors defined IDS as security systems that are responsible for the detection of any unauthorized behavior within the network caused by compromised nodes with network disruption intention. IDS targets to identify malicious frames that reside in the network as attackers and reduce misinterpretation of legitimate users as malicious ones [12]. Intrusion detections would provide all or some of the very useful information about the location of the intruder, intruder IP, intrusion type, time of the intrusion and the affected layer to the mitigation phase [8]. If the security technique is powerful enough, it will be able to identify the attackers soon enough to eliminate the severe damages to the network [13]. IDS are the fundamental infrastructures in security for wireless networks to provide detection of sophisticated and hard-to-find attacks in the wireless networks. The ultimate goal of the IDSs is to identify malicious activities due to the existence of the attacks. Although IDSs has advanced in recent years, they still suffer from a numerous number of attacks. IDS is the second line of defense. The reason behind it is that other cryptographic techniques deal with the external attacker nodes which are known as the second

line of defense and IDS is responsible for identifying the internal attacks. The necessity of Intrusion detection systems relies on the fact that conventional security techniques are not able to identify the internal compromised node that attacks the network and is only able to identify external attacker nodes.

In general, intrusion detection techniques can be categorized into five groups based on the methodology they adopt, namely: Anomaly-based IDS, Misuse-based IDS, Hybrid IDS as well as Hybrid Detection and Cross-layer IDS. Anomaly detection techniques focus on establishing a model of normal behavior of the network that can be captured based on the network monitoring in normal condition. In contrary, misuse-based detection uses predefined attack patterns and try to match the network traffic behavior pattern with previously known attacker models and raise an alert about intrusive attack whenever there is a match. Specification-based IDS is a mixture of the both above-mentioned techniques while cross-layer IDS aims to use multi-layer attributes to enhance the detection accuracy.

The objective in any classification algorithm is categorize data into separate classes. Using part of the whole data or in some cases using training data, a model based on previous data with known behavioral classes are built, called training model. The model is then used to classify new data called testing data. In other words, in the training phase, a set of data where each instance consists of couple of attributes and the corresponding class label is used to build a model and this model in further used in the testing phase where another, mostly new and previously unseen, set of data with similar features but may differ in value is classified based on their class labels.

Classification of Intrusion Detection Systems Based on the Detection Approach

A. Anomaly-based IDS

Anomaly-based IDS relies on the statistical behavioral modeling. In this technique, normal operations of the nodes are profiled. The detection of intrusions with this approach is based on the deviation of the network behavior from normal instances. The behavior of the network can be derived from the network traffic data. This approach relies on the historical data about the various behavior of the network. The derived data from the network will be compared with the predefined normal instance in the network and any deviation from normal network behavior would raise an alarm. Every anomaly-based detection algorithm consists of two phases. In training phase, a framework of the normal behavior of the network is created using a collection of the network behavior traffic features and traffic characteristics. In the testing phase, however, the ongoing network traffic is compared and testing with the training data to identify the nodes that do not follow the normal behavior pattern and mark them as intruders [5].

The main two assumptions in this detection scheme are that the number of attackers is less than the number of normal nodes and the intruder traffic pattern differs from the normal traffic pattern. Anomaly-based detection system derives the differences between network traffic flow and the predefined normal behavior and if the difference exceed some threshold, defined based on the sensitivity of the application for intrusion detection, it will raise an alarm about the existence of an attacker in the network. This technique can detect novel attacks. One major advantage of anomaly-based intrusion detection is in its capability of detecting previously unknown and new intrusions. It is a very interesting factor because in many environments not all kinds of attacks can be predefined and new vulnerabilities can be introduced to the network

and cause corruption in the network behavior. Some other advantages of anomaly-based IDS are that it is very useful in an insider attack. Moreover, they are examining traffic and looking for unusual or atypical behavior rather than looking for a particular pattern and as a result there is no longer a need to keep a signature dictionary up to date.

However, this technique suffers from a high false positive rate due to the following reasons. First, since collection of data over a period of time to derive normal network behavior pattern might contain intrusive attacks as well, it might define the intrusive network operations as normal behavior and mislead the detection accordingly. Moreover, the establishment of normal traffic records are done manually and any mistakes in defining the parameters that define the network normal behavior might lead to a misleading decision by the detection algorithm. Another major concern with this approach is that the normal profile needs to be updated frequently due to the rapid change in the network behavior [8]. The challenge in this approach is to how to define the normal network behavior due to the fact that the normal behavior of the network varies over time which can cause high false positive rate [4]. One way to address this issue is to establish the normality of the network behavior by automated training, as suggested in [9]. Anomaly detection can be categorized into the following techniques:

a. Statistical-Based IDS In this approach, a normal profile is generated based on the captured network traffic. After monitoring the network, the anomaly scores are generated based on the comparison with the normal references. The score difference represents if a network traffic instance is normal or abnormal. In order to identify compromised or abnormal instances, there required to be threshold which helps to distinguish normal activities form abnormal ones [17].

b. Knowledge-based IDS This approach relies on the prior knowledge about the network parameters in normal operating condition as well as under attack situations [17]. In rule-

based or knowledge-based approaches, an anomaly detector learns rules that describe normal behavior of the system. A network event that is not covered by the defined rules is considered anomalous.

c. Machine Learning-based IDS Data mining is designed to extract the pattern from any datasets. These pattern can be used to identify abnormal behavior of the network from normal ones. Therefore, they are useful for attack detection in communication networks. Machine learning is one of the approaches in data mining techniques. Machine learning can learn the pattern from the data and use the learned model to make predictions accordingly. In machine learning based IDS, a pattern is generated to analyze the inputs and distinguish the normality or abnormality of the traffic. Machine learning based IDS highly depends on the training set of instances [17]. Machine learning techniques are a broad range of identification and classification techniques that evolve detection models based on the input of previous data. Learning techniques are commonly used in the literature for anomaly detection purposes. Machine learning techniques can be further classified into supervised learning and unsupervised learning.

1. Supervised Learning: Supervised anomaly detection establishes the normal profile of the system through training using labeled data set. The labeling process is very costly, in terms of time consumption and probability of error in labeling.

2. Unsupervised Learning: In contrast to the supervised learning, in unsupervised learning, the detection does not rely on any prior knowledge of attacks or normal instances. In other words, the training data is unlabeled in unsupervised learning algorithms. These techniques can occur online due to its nature and has better accuracy comparing to supervise learning.

B. Misuse-based or Signature or rule-based IDS

Misuse-based detection model is also known as or signature-based IDS. Unlike anomaly de-

tection that generates a normal pattern of the network and compares all other network traffic behavior with what is known as normal behavior, misuse detection focus on the signature of the attacks. In other words, the system models the intrusions and attacks and store a signature of a set of known attacks and compare the network traffic data to find a match pattern to alert the network about the existence of an intruder. In misuse detection scheme, all the network traffic instances are compared with a predefined signature of previously known attacks. Despite its efficiency in detection of previously known attacks in the network, misuse-based IDS fails to new and novel attacks that the detection system does not have a signature of them.

In addition, another major shortcoming of this method relies on the fact that it is very difficult to define a signature of an attack with all its variations. In fact, the main concern about this technique is the limitation of the predefined signature of attacks which makes this technique to not be able to detect new unknown attacks and result in performance degradation. In other words, in this approach, predefined attack rules and signatures are used for detection purposes. Signature-based IDS works well with the known attack; however, it is unable to detect attack for which it has not a signature in its data base. Due to the existence of the predefined rules, it requires high computational power. Another major drawback of this approach is that the data base needs to be updated continuously.

C. Hybrid IDS

Hybrid IDS combines the advantages of both anomaly-based and misused-based intrusion detection techniques. It uses the predefined normal behavior of the network to com- pare the captured network traffic behavior and describe the correctness or the normality of the network operation based on the predefined instances. Since this detection relies on the manual specifications, it has low false positive rate and is capable of detection of the new and unknown attacks

[4]. This method is a combination of both misused and anomaly based intrusion detection systems. This approach has low false positive rate and is capable of identifying unknown attacks as well. In other words, it uses the advantages of both anomaly-based and misused-based IDS [17]. In this approach the deviation from correct operation of the network would raise an alarm however, these normal operations are set manually. It uses behavioral specifications to detect intrusions and depend on an expert to define a threshold for the system. This approach is capable of the detection of the unseen attacks.

D. Cross-layer IDS

Cross-layer IDS extract features from multiple layer of the protocol to detect intrusion in either a single layer or multiple layer of the protocol. Based on [10], if the IDS resides in one layer, as observed in a various number of publications, the other layers are prone and very vulnerable to the attacks. Cross-layer intrusion detection systems are extremely useful in determining the attack that takes place in more than one layer of the protocol stack.

In the layered structure, the routing path and data forwarding behavior is determined by network layer, the medium access and expected transmission time in link layer and power control and rate control in the physical layer. Therefore, in a layered structure, where there is no interaction among different layers, network layer, for instance, would not have access to any information about whether there is a congestion in the network, or if there exists a node with very low energy that cannot relay the transmitted message and hence results in the selection of poor routing path which may consist malicious nodes and as a result the information packets will be dropped due to other reasons than malicious activity in the network and the number of false positive would increase significantly. Independent security solutions in every layer might result in conflicts and degrade the performance. Therefore, to ensure the security and network

reliability, the security solution should cooperative coordination among different layers which would lead to robust intrusion detection systems [14]. Another issue in current IDS is that they fail to distinguish between faulty network and true intrusions in the network which increase false positive rate significantly.

Most of the single-layer intrusion detection schemes in the literature fail in distinguishing malicious activity from faulty network behavior due to legitimate reasons [14]. Traditionally, each layer of a protocol performed its tasks independently; however, in UDP protocol design, even though application layer can directly connect to network layer bypassing transport layer but with an inevitable delay which degrades the overall network performance. In this regard, unlike layered architecture where every layer is only allowed to communicate with the adjacent layers, cross-layer design can be of much help to significantly improve the performance due to using the cross-layer information. A more comprehensive study on cross-layer IDS and its design criteria is discussed in details in the fifth chapter.

Classification of Intrusion Detection Systems Based on the Place of Deployment

A. Host-based IDS

Host-based IDS is concerned with the events on the host that they are serving. They are capable of detecting the following intrusions: changes to critical system files on the host, repeated failure access attempts to the host, unusual process memory allocations, unusual CPU activity or I/O activity. HIDS achieves this by either monitoring the real-time system usage of the host or by examining log files on the host. HISDs are highly useful when an adversary tries to intrude and steal sensitive and confidential information stored on a particular host. These systems do not offer any form of defense for attacks against the other hosts in the same network.

As such, some of the widespread applications of such IDS systems are in mainframe computers, critical servers and common desktops.

B. Network-based IDS

Network-based IDS passively or actively listens to the network trans- missions, captures and examines packets that are being transmitted. NIDS can analyze an entire packet, payload within the packet, IP addresses or ports. Network-based IDSs work by studying the network traffic at different levels of abstraction. The sole aim of such systems is to study network traffic and to detect if there was an attack. If there was an attack, they either raise an alarm, create a log or respond to the attack, depending on the way in which the systems were configured

2.1.3 Security Mitigation Techniques

After detection techniques distinguish between normal behavior from abnormal ones and after the identification of the compromised nodes in the network, the network is entitled to remove the malicious node and secure the network. Mitigation strategies are the response or the reaction of the network to the attack and is the final step in security systems to the attackers.

In mitigation step the information provided by detection technique such as identification of the intruder, location of the intruder, the time of the intrusion, class of intrusion activity whether active or passive, intrusion type as well as target layer of the intrusion are used. The detected intrusions can be either bypassed such as revoking the adversary nodes from the network routing tables, ignored or even physically destroyed in more severe cases and applications. According to the security strategy. Applying Quality of Service (QoS) principals is one effective approach to mitigate detected attacks. QoS techniques can be used both on packets

and connection level. Resource allocation, for instance, is one commonly used techniques for attack mitigation, in which, less resources (e.g. power, bandwidth, access time, etc.) are allocated to the malicious nodes as to that of the legitimate nodes. Further study in this regard remains for future works.

2.2 Related Works and Suggested Studies

Intrusion detections are widely studied in the literature [15–20]. In [15], a survey of intrusion prevention and detection techniques is provided and various detection and prevention methods are classified and studied in details, including their advantages and drawbacks. Moreover, the efficiency and the challenges on cloud computing and etc. are discussed. Similarly, the authors in [16] and [17] surveyed various intrusion detection techniques. A taxonomy of intrusion detection systems and detection principle are presented in [18]. A comprehensive review on the learning and detection methods in IDS as well as a complete discussion on the problems with existing intrusion detection systems is presented in [19] and [20].

Chapter 3

Feature Selection in IDS

3.1 Introduction

The dramatic increase in the network traffic data has become a major concern in security systems. Intrusion detection systems (IDSs), as common widely used security systems for communication networks, are not an exception. An IDS monitors the network traffic to detect attacks through classifying the network traffic data into normal and abnormal classes. Due to the high dimensionality of the network traffic data, it is not always feasible for an IDS to detect intrusions quickly and accurately. Therefore, it is essential to derive only the informative features and attributes from the network traffic records that helps in intrusion detection and use them to measure the normality of the traffic flow.

In this regard, feature selection emerges as a necessary step in designing an IDS to overcome its shortcoming and enhance its performance through the reduction of its complexity and acceleration of the detection process. Features are the extracted from the network information packets that are sampled over a period of time and they describe the network traffic pattern.

Feature extraction and feature selection are one of the necessity of such systems due to the fact that they can directly affect the performance of the detection mechanism. Feature Selections are adopted to find the most informative attributes from the network traffic flow while removing irrelevant features and eliminating redundancies. Feature selection uses the correlation between network attributes to derive a small subset of the most informative attributes that represent the network traffic behavior for security purposes. Beside the above-mentioned advantages of feature selection in IDS, it accelerates the detection process since less number of network traffic features are adopted and used in detection.

To this end, in this study, we address the problem of dimensionality reduction by proposing an efficient feature selection algorithm that considers the correlation between a sub- set of features and the behavior class label. Correlation-based feature selection (CFS) and symmetrical uncertainty (SU) are the two correlation metrics used to measure the dependency level between features and class labels, and among features. Experimental results on NSL- KDD dataset shows that the proposed approach with fewer features, significantly outperforms the existing schemes in terms of the training time, time taken to build the model, while it preserves or increases the system accuracy. In addition, the efficiency of the proposed feature selection technique is tested on different classification algorithms and comparison results indicates that J48 classifier with the highest accuracy and precision values and lowest miss rate and false alarm rate values, performs better with the proposed feature selection technique.

Intrusion detection system (IDS) is a security method for monitoring and analyzing the network traffic patterns to find and report any abnormal behavior. A major concern in current IDSs is the high dimensionality of the network data so that the classifiers cannot discriminate the normal behavior of the system accurately and in a timely manner due to the existence of

irrelevant and redundant features. Thus, they suffer from poor prediction, high computational overheads and slow detection. On the other hand, due to the huge number of feature subsets that can be selected from input features, depending on the feature set dimensionality, it is challenging, if not impossible to use an exhaustive search and test each and every subset [67]. Therefore, the need for feature selection becomes inevitable.

Feature selection has been used as an efficient method to overcome the high dimensionality of real network traffic data for IDSs results in less training time, minimization of learning overhead and improvement in learning performance. Unlike feature extraction that introduces new features from a dataset, representing the network traffic data with a certain number of records and set of features that characterize the behavior of each record, the goal in feature subset selection is to choose the small set of the most informative features that characterize different network traffic behavior classes, efficiently and effectively [54]. Hence, feature selection leads to minimizing the complexity and computational time. To this end, features with limited information to predict the network behavior should be removed. Moreover, features with high correlation to one or more other features, called redundant features, should be eliminated. High correlation between two features indicates one has a portion or all of the information of the other features. Therefore, even though both features have valuable information, one is redundant and should be eliminated. Thus, redundancy reduction and irrelevant feature removal are the two main objectives in any feature selection algorithm.

In order to find the best subset of features, evaluation techniques are required to find the most important ones. Some techniques utilize machine learning algorithms, while others are based on the statistics of training data [88]. Although using learning algorithms to evaluate the importance of features results in higher values for accuracy, they are not efficient since they

consume more time to build the model and have complex computation. Furthermore, training the system with learning algorithms causes the whole system to become biased towards a particular classification algorithm [34]. In contrast, the filter methods that discriminate class labels based on statistical characteristics of the training set, are much faster than machine learning techniques while maintaining the same accuracy. Since the amount of captured network traffic data is immense and the filter method exhibits better performance when dealing with a large number of features, we address our proposed feature selection based on the filter method. Two main categories of filter-based feature selection methods are weighting algorithms, which select good features individually, and a subset search selection in which the best subset among all other subsets is selected based on evaluation metrics [54]. Our proposed method is based on the latter case because the weighting technique does not consider the possible interaction among features.

A group of carefully selected individual features does not necessarily outperform a good group of features [19]. This indicates that good individual features are not always the best fit in the feature subset selection because both linearity and non-linearity inter correlation among feature should be taken into considerations. This led us to present a two-stage filter-based feature selection algorithm to measure the dependency (relevance) of features to the class label and to other features[1]. Our proposed algorithm is based on the least linear or non-linear correlation[2] among features and highest correlation between features and the class label. In the first stage, correlation-based feature selection (CFS) is used to select a candidate subset of features with the maximum relevance to the target class label and minimum redundancy to the rest of

[1]Note that correlation and mutual information are two demonstrations of dependency or relevance [67].
[2]Throughout this study, the term "correlation" does not refer to any particular correlation measures.

features, while in the second stage, symmetrical uncertainty (SU) eliminates the features with high non-linear inter correlations. In other words, SU confirms whether candidate features are good fit in the subset of selected features.

We compare our method with other well-known existing feature selection techniques, including information gain, chi-square evaluator and gain ratio. In addition, different feature selection algorithms are tested with different classifiers and the performance of the selected features is measured with different metrics (e.g. accuracy, detection rate, training time). It is shown that with the selection of as few as the four best features based on the proposed method on NSL-KDD, the complexity and computational time of the system are enhanced significantly while preserving and improving in accuracy compared to other feature selection techniques as well as the whole dataset.

As discussed earlier, the need for feature selection techniques are vital since feature selections selects the most informative features and derive a subset of network traffic attributes that is free of irrelevant features. In addition, the redundant features are diminished in the selected subset. The selection of features would help to enhance the efficiency of intrusion detection since only the most informative features about the network traffic are used. Moreover, feature selection reduces the complexity and training time since only a subset of all features with less number of attributes from a whole dataset is used for detection of malicious behavior in the network. It is notable that with more number of features the accuracy becomes higher; however with the cost of computational complexity and very high required training time. However, if the number of contributing features in defining the secrecy of the network traffic behavior is selected wisely, one can reduce the training time to provide close to real-time training and decreasing the complexity. It has been proved that the reduction in number of features can de-

grade the performance since less number of features provide less amount of information about the network traffic behavior. On the other hand, if we select the right number of features, not only the intrusion detection become more efficient in terms of time consumption and complexity but also the detection accuracy can remain similar to using all the attributes or in our case, we could improve the detection performance slightly better.

3.2 Feature Selection Schemes

Different feature selection schemes based on dependency level have been discussed in the literature. Selection of features with minimum feature-feature redundancy and maximum feature-class relevance are the two fundamental basis of all these techniques.

Mutual information feature selection (MIFS) is a very well-known technique in which features based on the following criteria will create a subset.

$$f(f_i) = argmax\Big(I(C; f_i) - \beta \sum_{f_s \in S} I(f_s; f_i)\Big) \tag{3.1}$$

where C is the class label and f_i and f_s are features from input subset of features and selected subset, S, respectively. $I(X; Y)$ is the mutual information(MI) between X and Y and it represents the amount of knowledge one variable provides about the other. It is worth mentioning that zero MI means two variables are independent and positive MI shows their dependency. It

is defined as:

$$I(X;Y) = \int\int p(X,Y)\log_2\Big(\frac{p(X,Y)}{p(X)p(Y)}\Big)dx_n dy = \int\ldots\int p(x_1,\ldots,x_n,Y)\times \log_2\Big(\frac{p(x_1,\ldots x_n,Y)}{p(x_1,\ldots,x_n)p(Y)}\Big)dx_1\ldots dx_n dy \tag{3.2}$$

where $p(\cdot)$ is the probability density function (PDF) and $p(\cdot,\cdot)$ is the mutual information. It can also be defined as:

$$I(X;Y) = H(X) + H(Y) - H(X,Y) \tag{3.3}$$

where $H(\cdot)$ represents the entropy and is defined as:

$$H(X) = -\sum_i p(x)\log_2 p(x) \tag{3.4}$$

$$H(X,Y) = -\sum_j\sum_i p(y,x)\log_2 p(y,x) \tag{3.5}$$

In the equation (4.15), β is a manually tunable parameter and it is the major drawback of this method. Additionally, the term on the left side measures the relevance of a feature to characterizing the class label and the right-side term calculate the redundancy level between already selected features and these two terms are not compatible if the number of features in the selected subset increases, due to the fact that the terms do not grow equally [32]. Thus, a parameter-free feature selection method, minimum redundancy maximum relevance(mRMR) was proposed as a solution to compensate the limitations of the MIFS. The mRMR selects features based on the most relevance to the class label and the least redundancy in the subset

based on MI [67].

$$f(f_i) = I(C; f_i) - \frac{1}{|S|} \sum_{f_s \in S} I(f_s; f_i) \tag{3.6}$$

where $|S|$ is the number of features in the selected subset In [67], the author used mRMR to build a candidate subset for given classification algorithms and derived a more compact feature subset. Similarly, mRMR is used in [79] and the author re-weight the terms to make a balance between importance and similarity.

Another method to compensate the constraints of MIFS is normalized mutual information-based feature selection (NMIFS), presented in [32]. This method addresses the shortcoming of MIFS and mRMR by taking the average of the normalized mutual information as a new dependency metric [104].

$$G = argmax_{f_i \in F}\Big(I(C; f_i) - \frac{1}{|S|} \sum_{f_s \in S} \frac{I(f_s; f_i)}{min\{H(f_i), H(f_s)\}}\Big) \tag{3.7}$$

However, selected subset, based on NMIFS, still suffers from redundancies. Hence, in [97], the author proposed another criterion based on NMIFS, however, with a better balance between left and right terms of the relevance criteria. According to the author the features maximizing the following will be selected.

$$\Big(\frac{I(C; f_i)}{min\{H(C), H(f_i)\}} - \frac{1}{|S|} \sum_{f_s \in S} \frac{I(f_s; f_i)}{min\{H(f_i), H(f_s)\}}\Big) \tag{3.8}$$

On the other hand, an improved NMIFS is proposed in [104] based on the following:

$$G = I(C, f) - max\Big\{\frac{I(f; s)}{min\{H(f), H(C)\}}\Big\} \tag{3.9}$$

In [9], the author proposed another approach to overcome the limitations of the MIFS by presenting a flexible mutual information-based feature selection technique (FMIFS). The selection of the features is according to the following:

$$G = argmax_{f_i \in F}\Big(I(C; f_i) - \frac{1}{|S|}\sum_{f_s \in S}\frac{I(f_s; f_i)}{I(C; f_i)}\Big) \tag{3.10}$$

in which $f_i \in F$ and $f_s \in S$ are features from the initial input set and selected subset of features respectively. Features with positive values for G will be selected.

The major concern in the above-mentioned studies is that since the non-linearity correlation is calculated for all of the features from the input dataset, it may cause computational overheads. In addition, since incremental search, meaning that one feature is selected at a time, has been used in all of the above-mentioned feature selection techniques all the possible interaction between features has not been considered. Fast correlation-based filter (FCBF) is another feature selection method, which despite all the above-mentioned methods it uses SU as a correlation measure, and unlike mRMR uses a backward selection technique. A modified Kolmogorov-Smirnov (K-S) FCBF is presented in [88], in which symmetrical uncertainty (SU) is used to remove irrelevant features and a K-S test is used to remove redundant ones [105]. Similarly, in [86], the author proposed a four-step FAST subset selection algorithm in which SU is used to remove irrelevant features and eliminate redundancies. However, the major

concern in this study, is defining the threshold (θ) in the first step to select relevant features. CFS and SU are the two most widely used information theory measures in feature selection techniques that find the linearity and non-linearity correlation between two variables, respectively [10] [29]. Both measures are used in our proposed feature selection algorithm and will be further discussed in the following section.

In the first phase, training phase, the attributes that can accurately characterize the normal behavior from abnormal ones are identified using statistical or data mining techniques and are used to build a training model to be used in the testing phase. The first phase in any intrusion detection system is to train the system with a proper model of normal or abnormal behavior. In the testing phase, however, the system is capable of detecting intrusions based on the training model. These two phases are depicted in Fig. 3.1 in more detail. In both phases, since raw data is not compatible with some algorithms (e.g., SU cannot calculate continuous variables), preprocessing techniques (e.g., transformation and normalization) are required to prepare the captured data for the further processing. Next, in the training phase, the features contributing more in the behavior classification of the network traffic data are selected based on feature selection techniques, as shown in Fig. 3.1 in the dotted frame box. Then, the system is trained with desired classifiers considering only selected features. In the testing phase, the classifier distinguishes normal behavior from abnormal behavior with the help of the training model using only selected features from the testing set as an input, known as a decision making step. Ultimately, the performance of the overall detection system should meet the criterion with a certain threshold, which is application dependent, in order to be able to label network traffic records in the testing set, either "normal" or "intrusion". For cases that the desired criterion is not satisfied, the system should start over the detection procedure.

3.3 Proposed Feature Selection Model to Increase Performance Efficiency

3.3.1 Proposed Feature Selection Method

Feature selection as an essential part of any IDS can help make the process of training the model less complex and faster while preserving or even improving the overall performance of the system. Every feature selection technique starts with the subset generation, as shown in Fig.3.1. Feature subsets are built using subset generation techniques which are not within the scope of this research. Then, the most informative subset will be selected using a feature subset selection. The feature subset selection can be based on distance, divergence, consistency, classification or dependency. In this study, we use dependency as a measure for relevance. Furthermore, searching for the best subset will continue until the desired criterion is met. A combination of both mutual information and Pearson correlation coefficient is used in this study to find the best feature subset. The proposed feature selection method is based on two dependency measures, correlation based feature selection and symmetrical uncertainty.

Correlation-based Feature Selection Technique to Remove Redundant Features

In the first step, CFS [40] is used to keep relevant features while removing redundant ones. CFS evaluates the subset of features instead of individual attributes and is capable of determining the usefulness of attributes by considering both redundancy among features and relevancy between features and class label. CFS [41] is considered over MI, mainly because the calculation of the PDF, specifically the estimation of multivariate densities, in a high dimensional dataset

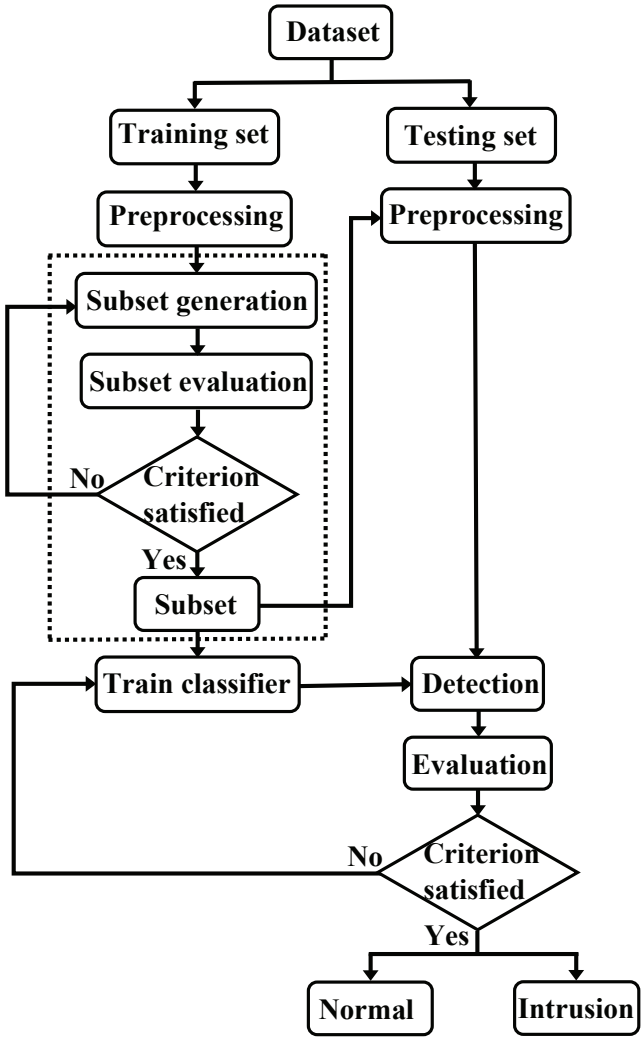

Figure 3.1: Intrusion detection system (IDS) model

increases computational time and complexity. Moreover, since mutual information is very sensitive to calculation of PDFs, any error in the estimation of them, drops the efficiency of the feature selection technique significantly [32]. The equation (3.11) indicates that the correlation between the whole subset and the class label is a function of the correlation between individual features with the class label and the inter correlation of features in the subset which can be

defined as follows:

$$CFS = r_{cs_m} = \Big(\frac{m\overline{r_{cf}}}{\sqrt{m + m(m-1)\overline{r_{f_i f_j}}}}\Big) \tag{3.11}$$

where m is the total number of features in subset s_m, r_{cs_m} is the total subset-class correlation based on feature-class and feature-feature correlation, $\overline{r_{cf}}$ is the average value of feature-class label correlation and $\overline{r_{f_i f_j}}$ is the average value of feature-feature inter correlation which are represented by the following equations, (3.12) and (3.13), respectively.

$$\overline{r_{cf}} = \Big(\frac{r_{cf_1} + r_{cf_2} + \cdots + r_{cf_m}}{m}\Big) \tag{3.12}$$

$$\overline{r_{f_i f_j}} = \Big(\frac{r_{f_1 f_2} + r_{f_1 f_3} + \cdots + r_{f_1 f_m}}{\frac{m(m-1)}{2}}\Big) \tag{3.13}$$

Consequently, (3.11) can be rewritten as:

$$CFS = \max_{S_m}\Big(\frac{r_{cf_1} + r_{cf_2} + \cdots + r_{cf_m}}{\sqrt{m + 2(r_{f_1 f_2} + \cdots + r_{f_m f_1})}}\Big) \tag{3.14}$$

where r_{xy} is the Pearson correlation coefficient between variables x and y and is defined as follows:

$$r_{xy} = corr(x; y) = \frac{\sum_{i=1}^{N}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{N}(x_i - \bar{x})^2 \sum_{i=1}^{N}(y_i - \bar{y})^2}} \tag{3.15}$$

in which $\bar{x}$ and $\bar{y}$ are the mean values of x and y, respectively. The correlation coefficient r_{xy} varies from -1 to +1 where values close to ±1 indicates two features are almost correlated and values around zero infers the features are uncorrelated. Since with an increase in the number of

features, the CFS will increase due to their low correlation [40], this method might select more number of features than is required to predict the class label. In addition, in real world communication, the dependency between network traffic data is not limited to the linear correlations. Hence, along with linear correlation (CFS in this study), non-linear correlation should also be considered and another metric with the capability of analyzing the non-linearity correlation is required. Thus, regardless of the type of the existing inter correlation among features and between features and class label, the system will be capable of finding characterizing features to the system behavior. Therefore, further processing is required to minimize the number of features in the candidate subset of features.

Symmetrical Uncertainty Technique to Remove Irrelevant Features

In the second step, symmetrical uncertainty is used to remove the features that do not collaborate with other features in the selected subset, in terms of the correlation to the class label. SU is defined as follows:

$$SU(f_j; f_z) = \frac{2I(f_j; f_z)}{H(f_j) + H(f_z)} \tag{3.16}$$

in which $I(x; y)$ and $H(x)$ can be calculated using the equation (3.3) and (3.4), respectively. As the equation (3.16) reveals, SU uses MI and entropy. Since it only calculates the mutual information between the selected number of features with corresponding class labels, it does not have the limitations of max-dependency scheme mentioned earlier [86]. Here, mutual information is chosen over other statistical correlation techniques due to the limitations of other statistical correlation techniques in determining the stochastically dependence and also they are unable to detect non-linear dependency [27]. Mutual information, itself, may not be an effec-

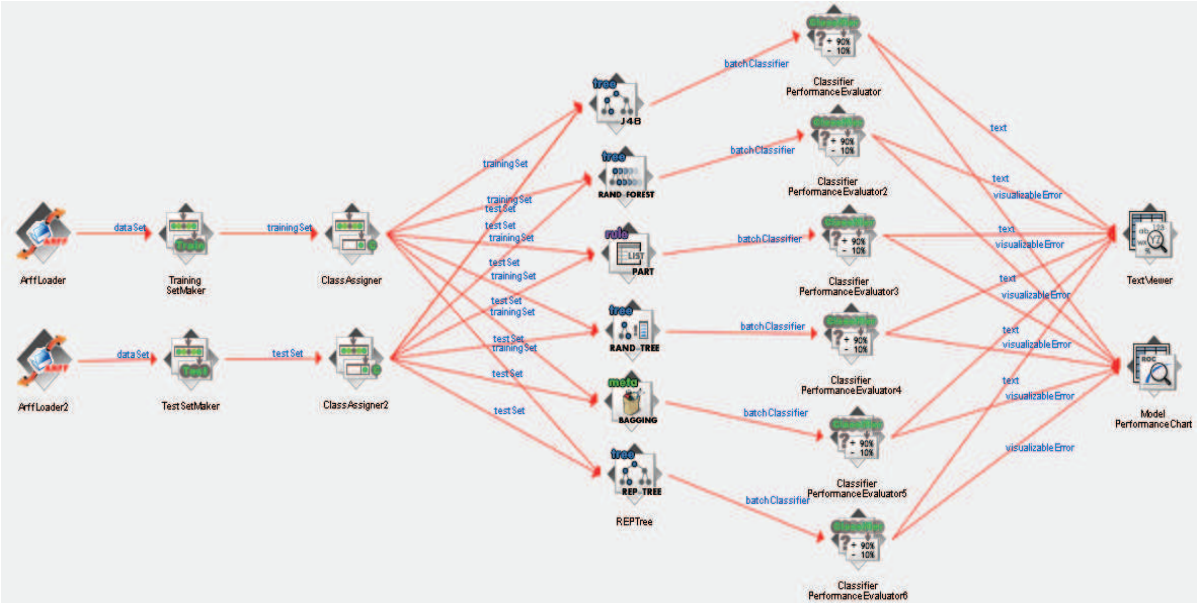

Figure 3.2: Testing phase of the intrusion detection system in WEKA

tive measure because they are biased toward the features which have greater values and neglect the features with the smaller values [29, 88]. Hence, symmetrical uncertainty overcomes the limitations of mutual information, by dividing it by the sum of the entropy of the features, while using its advantages. Also, unlike the mutual information, SU is a normalized metric and takes values between zero and one. Lower values for SU between two features implies the mutual knowledge of features from each other is less and hence the redundancy is less.

3.3.2 Proposed Feature Selection Algorithm

The procedure of the proposed feature selection technique is presented in Algorithm 1. where C indicates the class label, S_N contains all subset of features generated by feature subset selection and can be considered as the whole dataset, X is the selected subset of features based on CFS, Y contains the weighted elements of X based on SU and Z is consist of the accuracy values of the system removing features one by one, respectively. First, subsets of features are built

Algorithm 1 Pseudo code of proposed feature selection algorithm

Input: feature subsets, $S_N = \{s_1, s_2, \ldots, s_n\}$, and C is the class label
Output: best feature subset, s_{best}.
Step 1: ***Candidates selection: remove irrelevant and eliminate redundant features***
Begin
for $i = 1$ to N **do**
    calculate CFS_{s_i} and $CFS_{s_{i+1}}$
    if $CFS_{s_{i+1}} > CFS_{s_i}$ **then**
        empty X and append elements of s_{i+1} to X
        order elements of X in descending CFS value
    end if
end for
return X
Step 2: ***Confirmation of the selected candidates (further redundancy elimination)***
Begin
for $j = 1$ to M **do**
    for $z = 1$ to M and $z > j$ **do**
        $Y \leftarrow SU(f_j, f_z)$
        order elements of Y in descending SU value
    end for
end for
return SU_k: the element with the maximum value in Y
for $k = 1$ to M **do**
    find the corresponding j and z
    run the system with $(X - f_j)$ features
    $Z \leftarrow Acc_k$
end for
return Z
for $r = 1$ to M **do**
    find $Z_r = argmax(Z(r + 1) - Z(r))$
    if $argmax(Z(r\prime + 1) - Z(r\prime)) \neq argmax(Z(r + 1) - Z(r))$ **then** find the corresponding j and z associated with Z_r
        $s_{best} = X$ - f_j
    else
        stop further feature removal
    end if
end for
return s_{best}

with a forward selection technique that starts with an empty subset and adds features one by one. Second, CFS is calculated for every subset of features and is compared to the rest. The features with the highest CFS will be selected and make the subset of X. Due to the above-mentioned reasons, there still might be features that do not collaborate with the rest of the features in X. Therefore, in order to remove features which are not good fit in the candidate subset of features, firstly, SU is calculated for every two features in X. Then, the SU values will be stored and ordered descending in the new subset, Y. Next, backward elimination technique helps to remove features with least SU. In this stage, the feature elimination will continue until with elimination of more features results in significant decrease in accuracy. To this end, considering K as an index of the element Y with the maximum value in, the accuracy of the system is calculated for each feature removal from X and stored in Z. It is worth mentioning that the accuracy here is calculated with the same training set. Finally, the iteration of feature removal will stop when there can be more than one feature found that with its removal the accuracy decrease equally. Considering this approach, we were able to select the four best discriminative features from 41 features, in NSL-KDD.

3.4 Simulation Results and Performance Evaluation

In order to understand the effectiveness of the proposed feature selection technique in terms of the training time, it should be evaluated relatively with other methods. Hence, we compare our work with six other filter-based feature selection techniques such as information gain, gain ratio, chi-square and CFS in terms of the training time because according to the authors in [55] these attribute evaluators outperform other filter-based feature selection schemes. In

addition, the performance of each scheme is represented in terms of miss rate, false alarm rate, precision, and accuracy. Furthermore, the proposed feature selection is tested with different classifiers such as random forest, J48, PART, and C4.5. The computational time, the time to build the training model, highly depends on the system configuration [81]. The experiments were performed on a 64-bit Windows 10 operating system, with 8GB RAM and Intel Core-i5 at 3GHz. We used WEKA 3.7 [2] [39] as our data mining tool with a heap size of 100MB. Fig. 3.2 demonstrates the simulation of the proposed intrusion detection system model. Both training and testing phase, using proposed feature selection technique in WEKA. In our experiment, two separate sets are used for testing and training. The feature selection process and the training the model are performed on training set while the classification and intrusion detection use the test set. Cross-validation and percentage split of the data into training and testing sets are two other possible approaches to define the test set. However, the results from testing may be misleading due to utilizing the same dataset for both training and testing phases. Therefore, applying the separate training set and testing set allows us to obtain more reliable results.

In our experiment, two separate sets are used for testing and training. The feature selection process and the training the model are performed on training set while the classification and intrusion detection use the test set. Cross-validation and percentage split of the data into training and testing sets are two other possible approaches to define the test set. However, the results from testing may be misleading due to utilizing the same dataset for both training and testing phases. Therefore, applying the separate training set and testing set allows us to obtain more reliable results.

Table 3.1: Confusion Matrix considering abnormal classes as a target

	Prediction Class Label	
Actual Class Label	**Normal**	**Abnormal**
Normal	True Negative	False Positive
Abnormal	False Negative	True Positive

Table 3.2: Confusion Matrix considering normal classes as a target

	Prediction Class Label	
Actual Class Label	**Normal**	**Abnormal**
Normal	True Positive	False Negative
Abnormal	False Positive	True Negative

3.4.1 NSL-KDD Dataset

We used NSL-KDD dataset in our experiment. NSL-KDD [78] is a very common, publicly available dataset among researchers for IDS. Since NSL-KDD has labeled records for both training and testing dataset and our proposed feature selection technique is based on supervised learning algorithms, this makes it an appropriate option. NSL-KDD [89] is derived from KDDCUP 99 dataset with a fewer number of instances, however, with the same number of 42 features, including class labels. There are 125,973 instances in the NSL-KDD training dataset while the testing set has 82,332 records.The datasets contain a total of 24 training attack types, with an additional 14 types in the test data only. NSL-KDD [1] gives more realistic results than the original KDDCUP 99 due to the fact that it has no duplicate records so that classifiers cannot be biased towards the most frequent records. Also, the redundant records in this dataset have been reduced significantly [35]. The feature in this dataset is divided into four types of features, namely: Basic features, Traffic features, Content features, and Host-based features as follows:

1. Basic features are all attributes that are obtained directly from examining the TCP/IP

connection such as the session duration, protocol type and numbers of bytes from destination to source and vice versa;

2. Connection Based Traffic features are divided into connections made in a prior predetermined interval or over a number of connections with the first type focusing on connections having a common destination host as the target connection and the second looking at connections with the same service type as the target connection;

3. Content features help to collect information that might help detect unusual or anomalous events. Typical features include a count of failed attempts to log onto the network, number of times root was accessed or numbers of shell prompts;

4. Host Based attributes are related to the history of a particular host within a certain time period;

NSK-KDD has several advantages over original KDD data set. First, since it does not contain redundant records in the training set, the classifier is not biased towards the more frequent records. Secondly, the performance of the learners will not be biased by the methods with better detection rate on the frequent records due to not existence of the duplicate records. Thirdly, the number of network traffic behavior instances (records) are reasonable in both training and testing sets and therefore, there is no need to randomly select a small portion as training and rest as testing like cross-validation.

3.4.2 Evaluation Metrics

The performance of the proposed feature selection technique is tested with different evaluation metrics such as: Detection Rate (DR), Accuracy (Acc) also known as Correctly Classified instances (CC), Precision (Pr), False Alarm Rate (FAR) also known as False Positive Rate (FPR) and Miss Rate (MR). These metrics can be easily computed based on the confusion matrix. Rows and columns of this matrix represent the actual and prediction class of each instance respectively. The four components of the confusion matrix are: true positive (the number of instances predicted positive and they are actually positive), true negative (the number of instances which were predicted negative and they are actually negative), false positive (the number of instances which were predicted positive however they are actually negative) and false negative (the number of instances predicted negative but they are actually positive). Depending on considering to find normal or abnormal classes, confusion matrix can be defined differently in WEKA. The objective is to find abnormal classes and therefore, predicting an actual abnormal class as a abnormal one will be our true positive and predicting an actual normal class as a normal one will be our true negative, shown in Table 3.1.

In performance analysis of IDS, positive refers to the attack and negative data refers to the normal data. Moreover, the decision of the detection is either right or wrong which can be represented by true and false, respectively. Hence, there are four combination of the decisions as follows:

True Positive(TP): The classifier correctly classifies the intrusion as the intrusion.

True Negative(TN): The classifier correctly classifies the normal behavior as normal.

False Positive(FP): The classifier misclassified normal behavior as intrusion.

False Negative(FN): The classifier misclassified the intrusion as a normal.

Accuracy:Accuracy is defined as a measure of the correctness of an IDS which measures the number of detection failure.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{3.17}$$

Precision: This metric measures the percent of predicted intrusions versus total number of real intrusions. The goal of the detection technique is to achieve high precision which can be interpreted as less number of false alarm rate. and can be defined as where precision can obtain values between zero and one.

$$Precision = \frac{TP}{TP + FP} \tag{3.18}$$

Recall: Recall is covers the missing part from precision and is a complementary to recall metric. Therefore, classifiers with higher recall are desired. Recall is also known as detection rate. Similarly, the value of this metric also falls between zero and one.

$$Recall = \frac{TP}{TP + FN} \tag{3.19}$$

and Miss Rate (MR) is the complementary of recall evaluation metric and is defined as:

$$MR = \frac{FN}{FN + TP} \tag{3.20}$$

F-Measure: Since previously two metrics does not completely define the accuracy of an IDS, the importance of F-Measure comes into play. F-Measure is a mixture of both metrics

and is defined as the following formula. In other words, F-measure is the harmonic mean of precision and recall.

$$F - Measure = \frac{2}{\frac{1}{precision} + \frac{1}{recall}} \tag{3.21}$$

False Alarm Rate:

$$FAR = FPR = \frac{FP}{FP + TN} \tag{3.22}$$

ROC curve: The Receiver Operating Characteristics (ROC) analysis are used to visualize the relation between FP and TP. In ROC curve, the x-axis represents false alarm rate while y-axis accommodates intrusion detection rate(TP). Received Operating Characteristics (ROC) of the system is obtained in this study. The ROC curve visualizes the trade off between false positive rate and true positive rate in x-axis and y-axis, respectively. For cases where the number of normal instances and abnormal instances are not balanced, it becomes more essential to use ROC curve. When number of the class labels are not balanced, assuming 90% of instances are normal while the rest of 10% of the instances in dataset are abnormal, the accuracy, for instance, cannot be a good evaluation metric because if the classifier classifies all instances as normal then the accuracy will be 90% which seems to be a good percentage value for accuracy, while the performance of classifier is relatively poor [23]. Hence, the ROC curve is a good alternative performance evaluator in such cases. The area under the ROC curves is calculated for different classifiers in Table 3.5 and 3.6.

3.4.3 Experimental Results and Analysis

The training time (i.e. time required for the classifiers to build the training model) for several feature selection techniques, namely: CFS, chi-squared, IG and GR are compared with ran-

Table 3.3: The performance of feature selection techniques with different classifier in terms of detection rate

	Classification algorithms					
FST	**J48**	**RF**	**PART**	**RT**	**Bag**	**C4.5**
Full dataset	81.5	80.5	81.3	81.4	82.6	81.5
CFS	76.2	79.4	78.4	78.1	76.2	76.3
Proposed	**86.1**	**83.2**	**81.9**	**83.6**	**85.9**	**85.9**
chi-squared	78.8	80.6	80.9	79.1	81.2	83.8
IG	80.1	80.2	78.3	81.5	81	83.3
GR	76	76.6	78.7	77.5	76	75.8

Table 3.4: The performance of feature selection techniques with different classifier in terms of false positive rate

	Classification algorithms					
FST	**J48**	**RF**	**PART**	**RT**	**Bag**	**C4.5**
Full dataset	14.6	15.6	**14.9**	16	15.3	16.2
CFS	18.9	16.5	17.2	17.5	18.8	18.7
Proposed	**11.4**	**14.9**	15.9	**14.6**	**12.8**	**12.8**
chi-squared	16.8	15.5	15.1	16.5	16.2	14.2
IG	15.8	15.8	17.1	14.9	16.5	14.8
GR	19	18.6	16.9	18	18.9	19.1

dom forest classifier in Fig. 3.3 The experiments are carried out on random forest classifier in the testing phase since it has the longest training time relatively, which results in a better illustration of the enhancement of the proposed technique in the overall performance of intrusion detection. As observed, the proposed scheme outperforms the existing techniques with significantly less training time, less than 15 seconds. Since obtaining less training time is valuable only if the performance of the overall system is not degraded and the effectiveness of the system is not compromised, we compare the performance of our technique in terms of miss rate, false alarm rate, precision and accuracy with other schemes in Fig. 3.4. Comparing our

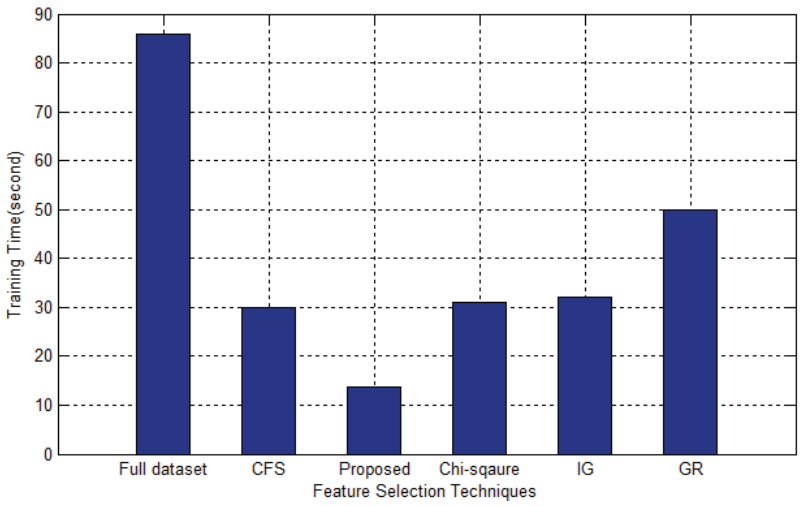

Figure 3.3: Training times for different feature selection techniques

proposed technique with 4 features against using the full dataset with 42 features, the figure indicates that not only no degradation is observed, but even some enhancement has been obtained. For instance, the miss rate and accuracy have been decreased around 3% and improved about 5% respectively. Moreover, it is shown that the proposed technique has the best performance among other feature selection techniques. Table 3.3 and 3.4 demonstrate the performance of different feature selection techniques tested with different classification algorithms, namely: J48, random forest, PART, random tree, bagging and C4.5 in terms of detection rate and false positive rate, respectively. As it can be observed, regardless of the classification algorithms, the performance of the proposed technique significantly improves comparing to other schemes. For instance, the detection rate of our proposed technique with 86.1% using J48 classifier outperforms the IG feature selection technique with detection rate of 80.1%. Similarly, Table 3.4 shows that the false alarm rate for the proposed technique with a significant drop comparing to other schemes helps to enhance the performance. For example, considering C4.5 classifier, the proposed scheme with 12.8% has significantly less false alarm rate than CFS and GR with 18.7% and 19.1%, respectively. It is notable that since separate sets are used for training and

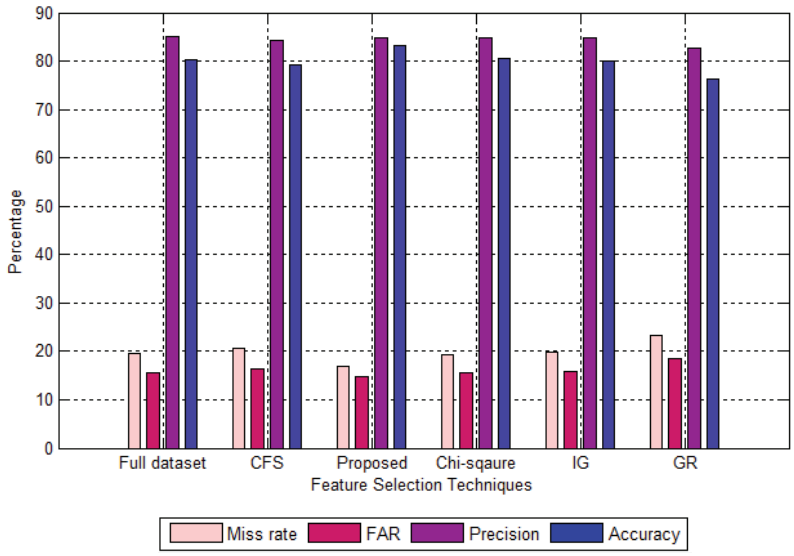

Figure 3.4: Performance comparison among different feature selection techniques

testing, the results are very accurate. However, on the other hand, when there is not different sets for training and testing, one uses cross validation technique to provide training and testing set from the whole single data set available. Using corss-validation technique has a drawback that the results based on it may vary a little for every simulation run. It is due to the fact that in cross-validation technique, the whole data set is divided into 10 parts as an example and only one part is required to be considered as the testing set and the rest 9 parts are considered as training set. However, since the selection of the testing part is arbitrary, cross validation technique, iterate until all the individual parts of the whole training set are used as the testing set. This may result in some variation in the results as different parts are used as the testing set in every iteration. However, this is not the case in this study.

On the other hand, the performance of different classification algorithms, for a system with the proposed selected feature and full dataset on NSL-KDD dataset are presented in Table 3.5 and Table 3.6, respectively. It is evident that, with respect to the significant decrease in the training time, the overall performance of the system using four features is enhanced comparing

Table 3.5: Classification results using only 4 features of NSL-KDD dataset

	Performance Metrics					
Classifiers	**MR**	**FAR**	**Acc**	**Pr**	**ROC**	**TT**
J48	13.9	11.4	86.1	88.2	91	3
Random forest	16.8	14.9	83.2	84.7	90	14
PART	18.1	15.9	81.8	83.8	90.7	7
Random tree	16.4	14.6	83.6	85	83.7	1
Bagging	14.1	12.8	85.9	86.8	91.6	5
C4.5	14.1	12.8	85.9	86.8	89.1	1e-4

Table 3.6: Classification results using all 42 features of NSL-KDD dataset

	Performance Metrics					
Classifiers	**MR**	**FAR**	**Acc**	**Pr**	**ROC**	**TT**
J48	18.5	14.6	81.5	85.8	84	36
Random forest	19.5	15.5	80.4	85.2	95.9	84
PART	18.7	14.9	81.3	85.6	81.7	49
Random tree	18.6	16	81.4	83.7	82.7	3
bagging	17.4	15.3	82.6	84.3	92.8	55
C4.5	18.5	16.2	81.5	83.5	82.2	8

to using 42 features, however with some exceptions. The false alarm rate for PART classifier, for instance, with an increase about 1% has a negative impact on the performance of using 4 features, however since the other evaluation metrics has increased or maintained at the same level, this adverse impact can be considered negligible. Moreover, among all classifiers, J48 classifier with the highest accuracy and precision and the lowest miss and false alarm rate, is considered to be the best classifier for the proposed feature selection technique.

3.5 Summary

In this study, a correlation-based feature selection technique for intrusion detection systems is proposed. We showed that in order to determining good features, along with considering their high correlation to class labels, and least inter correlations, the features should fit well in the selected subset and considering the inter correlation among features after the selection of the candidate subset is necessary and corresponding selected features are the best features contributing to discrimination of network behavior. Correlation-based feature selection(CFS) and symmetrical uncertainty(SU) are the two dependency metrics used in this study. The proposed feature selection technique is compared with other well-known feature selection algorithms namely: CFS, IG, GR and chi-squared on NSL-KDD dataset. The results indicate that the proposed technique has considerably less training time while maintaining accuracy and precision. In addition, different feature selection techniques are tested with different classifiers in terms of detection rate and FAR. Regardless of the classification algorithm, the results indicate that the proposed scheme out performs other techniques. Another observation from comparison results between the proposed technique and using the full dataset is that J48 classification algorithm performs better with proposed feature selection algorithm than other classifiers.

Chapter 4

Intrusion Detection Using SVM and RVM Classification Algorithms

4.1 Introduction

Broadcast nature of wireless communications has caused wireless networks to be very accessible to non-legitimate users. Moreover, the rapid growth of the computer network activities and the significant growth in the number of computers and electronic devices has increased the rate of network attacks and the intruders and hence, network security becomes even more crucial for wireless networks. Intrusions can be simply defined as a set of actions that compromise the confidentiality, integrity of the data and the availability of resources in the network [56]. Intrusion detection has been introduced as a second line of defense and a is very effective approach that involves monitoring of the network traffic pattern to detect malicious network behavior and prevent further unauthorized access, activity or any change of the data [70]. Intrusion detection system (IDS) analyses the network traffic patterns to find and report any abnormal behavior.

IDS was first introduced by [11].

IDS generally starts with preprocessing of the raw data and feature selection, in which it identifies and chooses the most informative attributes that characterize different network traffic behavior and have most of the contribution in differentiating different classes of the network traffic pattern. Feature selection is used to eliminate redundant features and remove irrelevant features for classification of the network traffic behavior into normal or abnormal classes [74].

After attribute selection, detection engine is the most important part of any IDSs that look for the abnormality in the wireless network behavior for which machine learning techniques have shown a great potential in distinguish normal network behavior from abnormal one providing accurate detection. The goal in machine learning techniques is to learn a model of dependency model of the class labels on the features with the objective of providing an accurate prediction or classification of a previously unknown instance, here, the network traffic instance [93]. However, a major concern and challenge in current IDSs is the effectiveness of the machine learning algorithm[1] that is used for abnormality behavior detection [53].

In the literature, many machine learning techniques have been discussed and compared and among all of them, Support Vector Machine (SVM) is one of the most popular machine learning algorithm used in the network traffic classification among researchers due to its good generalizations and classification accuracy. It is a supervised machine learning algorithm widely used in anomaly detection. SVM is based on statistical learning and can be easily defined as a discriminative classifier that segregates two class labels using a defined hyper-plane [70]. The fundamental concept of the SVM is to find a decision hyperplane[2] that can separate the classes

[1] It is notable that in this study we only considered supervised learning, where the class label of each instance is available in the training set

[2] In one dimensional space, the separation plane is called the a point while in two dimensional space, it is called a line; however, for more than two dimensional space hyperplane is the name of the separation plane [70]

effectively. SVM tries to separate the distinct classes by maximizing the gap between the hyperplanes that separate classes [69]. The goal is to have the hyperplane as far as possible from data points of each class. meaning that, the distance between the hyperplane and the parallel line to the hyperplane that crosses at least one of the instances from each class should be maximum. If the hyperplane has the same distance from the above-mentioned parallel lines, it is the optimal hyperplane and the total distance to these two lines, in two class scenario, is called the margin. In other words, the margin can be defined as the width that the boundary could be increased by before hitting a data point. The goal of the SVM is to train the model that assigns new unseen objects into a particular category.

Support vector machine creates a new feature space which is a finite-dimensional vector space in which each vector represents features of a particular object. This machine learning algorithm separates the input data into different classes based on different Kernel methods that map the data into a higher dimensional space. Much of the benefit of SVMs comes from the fact that they are not restricted to being linear classifiers. The kernel can introduce much more flexibility for non-linear decision boundaries. In order to avoid over fitting, sophisticated mathematical principles are applied and are going to be discussed further.

On the other hand, since most of the machine learning techniques provide a hard decision, either 0 or 1 for normal and abnormal behavior, respectively, which does not provide the uncertainty measure about the decision, it becomes significantly important to consider the techniques that provide the uncertainty about their prediction decision with utilizing the probability concept. Relevance Vector Machine(RVM) has attracted attentions due to its ability in providing probabilistic decisions. To this end, the performance of the RVM, as one of the very powerful machine learning techniques.

Since performance of the intrusion detection system highly depends on the efficiency and effectiveness of the classification algorithms, all contributing factors in performance enhancement should be taken into considerations. In order to elevate the performance of the SVM and RVM, their parameters need to be optimally tuned and hence, one major contribution of this study is to demonstrate the effect of the different parameters on the classification performance of these two algorithms. Moreover, to alleviate the low detection accuracy and high false alarm rate which are the shortcoming of current IDSs, in this study, we demonstrate a comparison analysis for different kernel techniques that can be used in SVM and studied the effect of various kernel tricks on NSL-KDD data set.

In this study, we first investigate the effect of different Kernel Methods and their parameters on SVM classification performance using NSL-KDD data set for intrusion detection purposes. Moreover, the performance of SVM and RVM for intrusion detection is compared in terms of detection performance, sparsity ability as well as generalization and effectiveness.

4.2 Support Vector Machine Technique for IDS

SVM is used to predict if the behavior of the network is normal or abnormal based on learning from the training data. We wish to categorized new unseen network behavior into two separate groups of normal and abnormal behaviors based on their properties and a set of known network behavior instances which have been already categorized. In other words, SVM works on the principle of fitting a boundary to a region of points which are all belong to a specific class label. Once the boundary is fitted on the training data, it can predict the class label of a new sample and determine which side of the built boundary it should reside.

In SVM, support vectors[3] are the data points from the training set which help in classification model construction. The vectors from the training data that are used for the prediction are called support vectors(SVs). These SVs reside at the boundary of the classification and closer to the separation line comparing to other vectors. After the training phase in which the classification model is built, the rest of data points are not going to be used and hence, SVM is a sparse model [63].Once SVM is trained using support vectors, the rest of the data points become redundant and therefore any small changes of the data cannot severely affect the hyperplane. In fact, one of the main advantages of the SVM is that the classification model only uses SVs and relies on the support vectors rather than the whole training dataset.

The objective in SVM is to find the optimal separating hyperplane since it can better classify the training data and can be generalized better with test data. The optimal separating hyperplane maximizes the margin of the training data. This hyperplane is called Maximal Margin Hyper plane (MMH). In the very simple cases, the linear hyperplane[4] can be found with the highest margin or distance from both classes. Margin space does not belong to any data point classes. It is notable that one of the main advantages of SVM is that classification accuracy has priority to the maximum margin. One of the key features of SVM is that the location of the MMH only depends on the support vectors which are the training instances that lies on the margin but not the hyperplane. In other words, the optimal classification occurs when such hyperplanes provide maximal distance to the nearest training points. In order to find the margin, the perpendicular distance from each training observation is computed for a given separating hyperplane and the smallest perpendicular distance from to a training instance is known as the

[3]The reason behind calling these data points as support vectors is that, they help the boundary to be created
[4]Hyperplane is a generic name of the boundary in more than three dimensional spaces

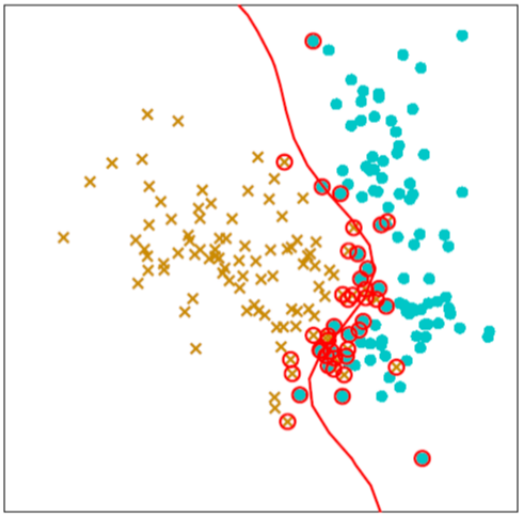

Figure 4.1: Support Vector Machine Classifier Demonstration

margin. MMH guarantees that it is the farthest minimum distance to a training instance. In more complicated scenarios where it is difficult to use a straight line to segregate, the data is not linearly separable, SVM introduces new features to the feature set and then use the linear hyper plane. SVM maps the input data into a much higher dimensional[5] feature space, where the separation decision surface can be found, using non-linear mapping in order to be able to construct a decision surface that can maximize the margin between two classes and separate them into two categories effectively. There are functions which take low dimensional input space and transform it to higher dimensional space and are very useful in non-linear separation problems. In other words, in order not to introduce new features manually, SVM uses a technique called Kernel which is a complex data transformation technique that provides new features in order to make the problem linear, hence, it is easier to be solved. It is notable that SVM classification accuracy highly depends on the kernel parameters. Using SVM to separate normal classes from abnormal ones, it is essential to tune parameters carefully since they can

[5]In SVM, each dimension represents a feature from the feature space and support vectors are instances from the training set

effectively improve the model performance or has an adverse effect on classification if they are not tuned perfectly.

The classification of SVM not only depends on the selection of kernel functions but also the performance of the SVM also highly relies on the training data, however, using the whole data is not efficient. Therefore, SVM requires feature selection in advance on the training data so that amount of information on the support vectors will decrease relatively and therefore, the computation complexity and detection accuracy will enhance significantly [70].

Moreover, the key feature of Support Vector Machine(SVM) is that it tries to minimize the error measurement in the training set and also maximize the margin between two classes simultaneously which is an effective strategy to avoid over fitting[6]. This helps to have a good generalization and results in to have a sparse model highly dependent to a subset of kernel functions. In other words, the sparse model is highly dependent to the support vectors which are the examples in the training set that resides on the margin [93].

In addition, Support vector machine algorithm is memory efficient which is an interesting capability for low memory sensors in wireless sensor networks. Additionally, this classification algorithm is able to learn both simple and highly complex classification models. Another interesting advantage of SVM comparing to other statistical methods are that it requires much fewer samples than the number of variables. Moreover, SVM is known for its robustness(is robust to a very large number of variables and small samples), efficiency and sparseness [106].

There are different types of SVM discussed in the literature. Linear SVM is commonly used since it learns faster and is able to classify new instances as compared to the non-linear SVM.

[6]It is notable that when a classifier learns to correctly predict output from given inputs in previously seen samples but fails to do so in previously unseen samples, it is over fitted. In other words, over fitting basically indicates that the MMH is a very good fit for the training instances however its performance is quite poor when exposed to the testing instances.

One-Class SVM [107] is used for unsupervised learning while C-SVM is used for supervised learning [56]. In [37], the author used Wavelet kernel LS-SVM for anomaly detection while Genetic SVM has been studied in [57]. An efficient SVM has been proposed to handle Port scan attacks in [96] to detect port scan attacks. The author proposed SVM with Bat algorithm for anomaly detection purposes [31]. In addition, an optimization method for parameters of the SVM is provided for network intrusion detection in [102]. The performance of SVM is compared with the combination of SVM and other classification techniques [26]. Although, this might provide better performance in terms of accuracy, sensitivity and specificity; however, the complexity becomes an issue. Moreover, if one selects the optimum value for the parameters that have significant effect on the performance of SVM, similar classification performance can be achieved without sacrificing the computational overhead. Therefore, the goal is to gain better classification performance with tuning SVM parameters instead of combining it with other classifiers. Support Vector Machine can be formulated as follows: Decision function

$$f(x) = \vec{w} \cdot \vec{x_j} + b = 0 \tag{4.1}$$

where $\vec{w}$ is slope of the hyperplane , $\vec{x_i}$ are the feature values and b is the y-intercept of the hyperplane and $f(x)$ is the classification function. In (4.1), the two dimensional scenario is assumed. Since this study is based on a two-class dataset, the hyperplane that efficiently separates two classes resides between the two hyperplanes with the exact same distance from the existing hyperplanes.

$$D = \frac{|b_1 - b_2|}{\| \vec{w} \|} \tag{4.2}$$

Where D is the distance between two parallel hyperplanes. In order to optimize the hyperplane margin in SVM, we want to maximize the gap between data points on the boundaries called support vectors. In other words, we need to maximize the distance, (4.2). The linear primal formulation objective function can be defined as maximizing the following:

$$D = \frac{2}{\| \vec{w} \|} \tag{4.3}$$

$$s.t \begin{cases} y_i(\vec{w} .\vec{x_j} + b) - 1 \geqslant 0 \; for \; y_i = +1 \\ \\ y_i(\vec{w} .\vec{x_j} + b) - 1 \leqslant 0 \; for \; y_i = -1 \end{cases} \tag{4.4}$$

where $i = 1, 2, .., N$ and N is the number of samples or instances. Equivalently, it can be rewritten as:

$$min \frac{1}{2} \|W_i\|^2 \tag{4.5}$$

$$s.t \;\; y_i((\vec{w} \cdot \vec{x_j}) + b) - 1 \geqslant 0 \tag{4.6}$$

The above-mentioned optimization problem is called primal formulation of linear SVMs and is a Quadratic Programming (QP) with convex objective function and subject to linear constraint with n variables which is the number of the features in dataset. QP optimization problems can be easily and efficiently solved by greedy algorithms since every local minimum is a global minimum. The solution of the quadratic optimization problem involves constructing a dual problem where Lagrange multiplier α_i is associated with every constraint in the primary problem. QP algorithms can identify which training points i.e. x_i are the support vectors with non-zero Lagrangian multiplier α_i. In linear dual formulation case, the objective function is to

maximize the following: For inseparable issues the following optimization problem should be solved, where ξ_i is the slack variable and C is the regularization constant to control over fitting.

$$min\frac{1}{2}\|W_i\|^2 + C\sum_{i=1}^{N}(\xi_i + \xi_i^*) \tag{4.7}$$

$$s.t. \quad ((\vec{w} \cdot \vec{x_j}) + b) - y_i \geqslant \epsilon + \xi_i$$

$$y_i - ((\vec{w} \cdot \vec{x_j}) - b) \geqslant \epsilon + \xi_i$$

$$\xi_i, \xi_i^* \geqslant 0 \tag{4.8}$$

Where ξ_i is used to relax the hard margin and C is used to manage the trade-off between classification error and maximal marginal of separation. The above formulation can be translated into the equivalent dual optimization problem:

$$\sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i,j=1}^{N}\alpha_i\alpha_j y_i y_j \vec{x_i} \,.\vec{x_j} \tag{4.9}$$

$$s.t. \quad 0 \leqslant \alpha_i \leqslant C \;\; and \;\; \sum_{i=1}^{N}\alpha_i y_i = 0 \tag{4.10}$$

In the above equations, $\vec{x_i}$ contains the feature values and y is the class label. Parameter C in the soft-margin SVM is the trade off between maximizing the margin and minimizing the classification error and w-vector will get a very small norm with the expense of misclassification. Optimal classification occurs when hyperplanes provide maximal distance to the nearest training data points. It is computational efficient on large datasets. It is also memory efficient since only subset of training set are used in the actual decision process of assigning new members, only these points need to be stored in memory. SVM are fast and scalable however, the need to

identify the right kernel function is very challenging. Therefore, the vector w is defined as:

$$\vec{w} = \sum_{i=1}^{N} \alpha_i y_i \vec{x_i} \tag{4.11}$$

$$b = y_i - \vec{w}^T \cdot \vec{x_i} \tag{4.12}$$

Then the solution becomes: To address the non-linear classification, kernel function is used and therefore, the output of the classification follows:

$$f(\vec{x_i}) = sign(\sum_{i=1}^{N} \alpha_i y_i K({x_i}^T \cdot \vec{x_j}) + b) \tag{4.13}$$

This basically means that, given the new instances x, SVM classifies and make decisions based on $f(\vec{x_i})$. It is notable that w is a vector with n elements and α is a vector with N elements which represent the number of features and number of samples, respectively.

For the cases of having noisy instances, all data points cannot be classified correctly. Therefore, slack variable ξ_i can be added to allow misclassification of difficult or noisy instances. Hence, the optimization problem can be reformulated as follows:

4.3 Kernel Functions and Their Affect on SVM

Kernel function reflects the similarity of two data points and can be interpreted as a similarity measure. As discussed earlier, in the cases where the linear decision surface does not exist and the data is not linearly separable, SVM uses Kernel function to project the data into a richer feature space and construct a hyperplane in that space. Kernels computes the dot products in

some feature space without knowing what this feature space is. Basically the equations are all the same however instead of x, we now use the mapped version of it which is Φ_x. Therefore, for instance, instead of equation (4.13) we can rewrite it as:

$$f(\vec{x_i}) = \text{sgn}(\sum_{i=1}^{N} \alpha_i y_i \Phi(\vec{x_i}).\Phi(\vec{x_j}) + b) = \text{sgn}(\sum_{i=1}^{N} \alpha_i y_i K(\vec{x_i}, \vec{x_j}) + b) \tag{4.14}$$

There are different types of Kernel functions that helps solving non-linear classifications using SVM. In the following kernel functions, x_i and x_j are two variables from feature space. It is very challenging to find the appropriate kernel function in data sets with large number of features.

Despite the effectiveness of kernel techniques in many applications, SVM demonstrate different performance with different kernel functions [85]. The proper choice of the kernel functions and the relative parameters can affect the performance of SVM significantly. The most popular kernel functions are polynomial kernel, RBF or Gaussian kernel, Sigmoid kernel as well as Linear kernel. RBF, in particular, has strong adaptation and convergence ability in network intrusion detection applications [85].

Linear Kernel:

$$K(\vec{x_i}, \vec{x_j}) = \vec{x_i}.\vec{x_j} \tag{4.15}$$

Gaussian Kernel: known as Radial Basis Function (RBF) Kernel

$$K(\vec{x_i}, \vec{x_j}) = \exp(-\gamma \parallel \vec{x_i}, \vec{x_j} \parallel^2) \text{ while } \gamma = \frac{1}{2\sigma^2} \tag{4.16}$$

In Gaussian Kernel mapping, geometrically, there are bumps or cavities centered at the training

Table 4.1: Effect of Gamma Using RBF kernel function in SVM on NSL-KDD dataset

γ	**no. of SVs**	**Training Time**	**Testing Time**	**RMSE**	**TP rate**	**F-Measure**	**ROC**
0.01	8266	7100	132	50.16	74.8	74.6	77
0.05	5876	9993.59	85.52	48.71	76.3	76.1	78.3
0.1	5043	8661	68.36	48.68	76.3	76.2	78.3
0.2	4905	4753	68.96	48.41	76.6	76.4	78.5
0.3	4557	5073	63.14	47.83	77.1	77	79
0.4	4417	4411	51.35	47.7	77.2	77.1	79.1
0.5	**4380**	**4456**	53.53	47.61	77.3	77.2	79.2
0.6	4472	4843	54.76	47.58	77.4	77.3	79.3
0.7	4573	5335	5587	47.59	77.4	77.3	79.3
0.8	4689	5544	68.95	47.48	77.5	77.3	79.4

data points. In other words, the resulting mapping function is a combination of bumps and cavities. RBF kernel uses Euclidean distance. Since RBF kernel decreases with the distance and ranges between zero and one, it can be considered as a similarity measure. In table 4.1, the performance of SVM with RBF kernel function with different values for Gamma is studied. From the table, it can be observed that classification performance of SVM using RBF kernel is not highly dependent to γ and remains almost the same with variation of the value of γ. However, γ highly affects the efficiency of the SVM. Table 4.1 shows that $\gamma = 0.5$ provides the highest efficiency in terms of time required to train the model. Since SVM classification algorithm using RBF kernel with $\gamma = 0.5$ requires fewer number of support vectors from the training set to build the classification model, it therefore, requires less time to train the model[7].

Polynomial Kernel: Polynomial kernel uses both given features and a combination of the the input features and similarities.

$$K(\vec{x_i}, \vec{x_j}) = (p + \vec{x_i} . \vec{x_j})^q \tag{4.17}$$

[7]Training time is highly dependent to the number of required vectors from the training set to build the classification model

where x_i and x_j are the vector of features computed from training and testing samples; respectively while p is the parameter that trade off the influence of higher-order terms versus the lower-order terms and q is the polynomial degree.

Sigmoid:

$$K(\vec{x_i}, \vec{x_j}) = \tanh(k\vec{x_i}.\vec{x_j} - \delta) \tag{4.18}$$

The performance of support vector machine detection algorithm with different kernel techniques is shown in Table 4.2. As it can be observed from the table, Gaussian kernel function performs the best comparing to linear, polynomial and sigmoid kernel functions in terms of accuracy, precision and F-measure evaluation metrics. Moreover, the area under the curve is higher in SVM with Gaussian kernel than other kernel techniques.

4.4 Relevance Vector Machine Technique for IDS

Relevance Vector Machine is a special form of the sparse kernel model which is based on the Bayesian framework and very similar to Support Vector Machine. RVM tries to adapt the target conditional distribution function, likelihood function as well as the link function to form the changes in the target quantities [106]. RVM is a special case of a sparse kernel function model which represents a Bayesian treatment of a generalized linear model of the similar function form to SVM. The difference relies on the solution. RVM provides the probabilistic interpretation of the outputs. Fig. 4.2 illustrates the classification of two dimensional feature space based using relevance vector machine with two class labels. The instances with red circle around them shows the relevant vectors and the red line shows the separating line of the two class labels. RVM have comparable performance to SVM in classification accuracy however,

Table 4.2: Effect of Kernel Functions on SVM Without Feature Selection on NSL-KDD Dataset

Kernel Function	Accuracy	Preciasion	F-measure	ROC
Linear Kernel	64.3	65	61.4	60.7
Guassian Kernel	**72.3**	**81.5**	**71.5**	**75.4**
Polynomial Kernel	56	32.4	41.3	50
Sigmoidal Kernel	53	56.7	52.5	55.2

the sparsity, generalization ability as well as decision speed should be studied. RVM can overcome the disadvantages that are associated with SVM The most interesting feature of RVM is that it requires fewer number of kernel functions [77]. RVM uses regression estimation to obtain the distribution of the prediction values and provide a sparse solution based on kernel functions [47]. RVM can easily handle the shortcoming associated with the SVM model and elude the complexity by producing models that have both structure and parametrization process which together, provide a better representation of the information content of the data. In RVM, firstly, an independent Gaussian prior is used and in the second level Gamma hyper prior is used for the variance parameters [95]. RVM requires more training time comparing to SVM since it computes the prior information for prediction of the class relationship. However, the prediction time is almost similar in both SVM and RVM. This classification model is more robust with less number of basis functions(RVs) [69]. Relevance Vector Machine(RVM) was first proposed by [93] and is based on Sparse Bayesian Learning(SBL). RVM leverage from requiring fewer number of relevance vectors and has better generalization ability than SVM [47]. RVM algorithm, like any other machine learning algorithm, falls into two phases: training phase in which the classifier is trained with the labeled feature, here we assume a supervised learning, and the testing phase which is also known as classification phase of the new instances [58, 77].

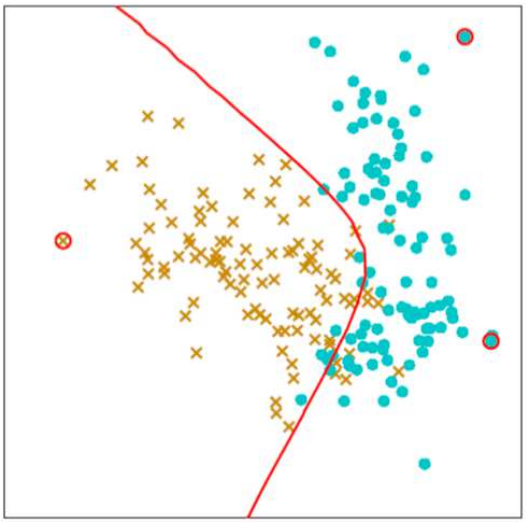

Figure 4.2: Relevance Vector Machine Classifier Demonstration

In this study, we applied RVM classification algorithm to assess the security of the computer network attacks and predict the normality or abnormality of the network behavior. SVM classification result are obtained through quadratic optimization while RVM is based on Bayesian framework [77].

4.4.1 Training Phase of the Learning Process

We assume x_i are the feature set while $1 < i < N$ and t_{ri} are the class labels. In intrusion detection as a system, we want to implement the relevance vector machine as a learning algorithm, we only have two classes, either 1 or 0 which indicates the existence of an intrusive node or a secure network system; respectively. RVM classification function is defined as the following.

$$y(x; w) = \sum_{i=1}^{N} w_i K(x, x_i) + w_0 \tag{4.19}$$

where $K(x, x_i)$ is the kernel function which defines one basis function for each example in the training set and W is a vector consisting of the weights of the model and is defined by (4.23). If we assume the probability of the occurrence of any of the classes follow a normal distribution, we have:

$$\mathcal{N}(x|\mu, \sigma^2) = \frac{1}{\sqrt{2\pi\sigma}}\exp(-\frac{x^2}{2\sigma^2}) \tag{4.20}$$

Also, if we consider each of the class labels are independent random variable, we got:

$$p(t|W, \sigma^2) = \prod_{i=1}^{N} \mathcal{N}(t_i|y(X_i; W), \sigma^2) = (2\pi\sigma^2)^{-\frac{N}{2}} exp(-\frac{\|t - \Phi\|}{2\sigma^2}) \tag{4.21}$$

where t, W and Φ can be defined as below:

$$t = (t_1, t_2, ..., t_N)^T \tag{4.22}$$

$$W = [w_0, w_1, w_2, ..., w_N]^T \tag{4.23}$$

$$\Phi = \begin{bmatrix} 1 & K(x_1, x_1) & K(x_1, x_2) & \dots & K(x_1, x_N) \\ 1 & K(x_2, x_1) & K(x_2, x_2) & \dots & K(x_2, x_N) \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 1 & K(x_N, x_1) & K(x_N, x_2) & \dots & K(x_N, x_N) \end{bmatrix} \tag{4.24}$$

Since there are many parameters in the training model, the estimation of maximum likelihood for w and σ^2, it will suffer from sever over-fitting.

4.4.2 Classification Phase of the Testing Process

In the testing phase, new class labels t_* are predicted based on the previously learned class labels t. Therefore, the probability of new class labels given previously known class labels can be find as follows:

$$p(t_*|t) = \int p(t_*|w,\sigma^2)p(w,\sigma^2|t)dwd\sigma^2 = \int p(t_*|w,\sigma^2)\frac{p(t|w,\sigma^2)p(w,\sigma^2)}{p(2)}dwd\sigma^2 \quad (4.25)$$

Appendix C provide a detailed description on how to classify new instanced in RVM and elaborates the solution to 4.25.

4.5 Performance Comparison Between Support Vector Machine Technique and Relevance Vector Machine Technique

In the literature, there are some studies that focused on the performance comparison between SVM and RVM [69,77,101,106]. The author compared the two classification algorithms based on their application to uncertainty analysis in [77]. Relative Mean Square Error is used as the metric for comparison and simulation results illustrated that the uncertainty will decrease with increase in the sampling size. The authors in [101] investigate the comparison between two classification techniques in terms of sparse property, generalization ability and decision speed; however, in text detection application. On the other hand, [69] and [106] compared SVM and RVM in terms of efficiency and effectiveness, respectively. SVM has high generalization

capability. It can also reach the global optimum solution while having an excellent classification accuracy and have a good performance overall [85]. SVM is a better classifier due to the following reasons [70]:

1. The learned classifier is based on the number of support vectors instead of the high dimensional of the data and therefore, SVM is less vulnerable to over-fitting problem;

2. SVM has a good generalization ability because of the few numbers of support vectors;

3. SVM is very effective and efficient and robust and makes accurate decisions [28];

4. Combination of kernel functions with SVM increase the generalization capability and help SVM to find the global optimum solution [28];

5. SVM relies on SVs for building the model therefore the size of the training data is not an issue [56] and are insensitive to data dimension [28];

Despite SVM advantages, there are some limitations and drawbacks that SVM suffers from. First, the computational complexity increases significantly with the increase in the number of the samples in the training set, due to the fact that the number of the support vectors grows with the size of the training set. Secondly, the predictions in SVM are 'hard' binary decisions and it is not based on the probabilistic prediction[8]. In addition, in order to obtain kernel parameters and the regularization coefficient, as SVM requires, the estimation of the error/margin trade-off parameter "C" requires cross validation techniques which cause more complexity in the training model and is wasteful in terms of computation and data. [93]. Moreover, the number of basis functions grows linearly with the growth in the training sample size which limits the

[8]Probabilistic prediction provides the uncertainness in the prediction decision

sparse model [47]. It is also notable that another drawback of SVM is that the MMH and thus the classification performance is significantly sensitive to the support vectors location. Apart from the above-mentioned disadvantages, SVM can only support numeric data and all other types of data require to be converted into numbers and also, if the normalization preprocessing technique will be adopted on the training dataset, it will fasten the speed of the SVM [59]. SVM is also very sensitive to the noise and even a relatively small number of mislabeled instances can dramatically degrade the classification performance. Although SVM is known for its well detection accuracy however, there are some drawbacks with this classifications algorithm [48]:

1. It provides a black-and-white and sharp decision about the predicted class labels, however, a probabilistic prediction is a more desirable method;

2. The number of support vectors, that are required from training set for the prediction of the class labels of the testing set, grow rapidly with growth in the size of the training set [101];

3. In SVM, kernel functions must satisfy Mercer's condition;

4. The estimation of the parameter C(i.e. trade-off between error and margin) is waste of data and computation [106];

5. There is no clear solution to determine the value of constant C and kernel parameters for non-linear kernel [63];

The main difference between these two machine learning technique is that in RVM the kernel function does not necessarily need to satisfy the Mercer's condition which requires ϕ to be a continuous symmetric kernel of a positive integral operator. RVM does not require to use

Mercer function instead it uses arbitrary basis function [93]. The two classifiers are basically from different principles. In SVM, the support vectors are defined by the hyperplane or known as border line which are the samples from the training set that are not easy to be classified and are located very close to the decision boundary of the classifier. However, in RVM, the samples that are relatively more appealing to be the representative of the two classes. These samples are located far from the decision boundaries of the classifier [95]. Unlike RVM, SVM is a non-probabilistic learning algorithm since the features in the new instances fully determine its location in the feature space and there is no stochastic element involved. However, much of the benefit of SVM comes from the fact that it is not restricted to being linear classifier and with utilizing Kernel trick they become more flexible by introducing various types of non-linear decision boundaries.

Based on the characteristics of the SVM and RVM, relevance vector machine is a better classifier in several aspects: first, since the number of the relevance vectors are much less than those of support vectors in SVM, RVM is sparser comparatively. Moreover, RVM uses a sparse prior probability while sparsity of the SVM comes from having support vectors a subset of training set however since with the increase in the size of the training set, the number of support vectors increases linearly, it can be concluded that RVM is sparser than SVM.

Second, in RVM, there is no regularization parameter C to be tuned as there is in SVM. SVM can control over fitting by soft margin approach discussed earlier. Since SVM is a simple convex optimization problem, it guarantees to converge to a single global solution. It is notable that RVM has its own drawback which is, in the training phase, RVM is involved in the highly non-linear optimization process. However, if we consider having the classifier be trained off-line, this issue is not of a great concern [95]. It is notable that SVM requires parame-

ter selection while RVM does not have parameter limitation. RVM is based on Sparse Bayesian Learning(SL) and SVM basis relies on the rule of structure risk minimum (SRM) [101, 103].

In this study, we are interested in comparing SVM and RVM based on sparsity ability, generalization ability as well as classification accuracy. Computation complexity of SVM is much greater than RVM due to higher number of samples of instances required for building the classification model [77]. Despite all the advantages of RVM, this algorithm also suffers from several disadvantages. One major issue with RVM is that it may reach the local maximum while in SVM it is guaranteed to reach the global optimum due to the fact that the optimization function is convex. High memory requirement and computational complexity are other drawbacks of RVM [63].

4.5.1 Performance Comparison in Terms of Sparsity

To measure the sparsity ability of SVM and RVM, we tested both classifiers on different portion of training set and derived the number of required vectors from the training set to build the model. SVM and RVM are tested on NSL-KDD dataset with 125973 number of samples in training set and 22544 number of instances in the testing set. On training phase, the number of support vectors needed for SVM with various different kernel functions and the number of relevant vectors required for RVM to build the model are shown on figure 4.3. Based on this figure, RVM is sparser than SVM due to the fewer required number of data points for classification. In other words, the number of relevant vectors (RVs) is much fewer than support vectors (SVs). From Fig 4.3 it can also be observed that with the grown in the training size, the rate of grown in number of RVs and SVs are not the same. With the growth in the training

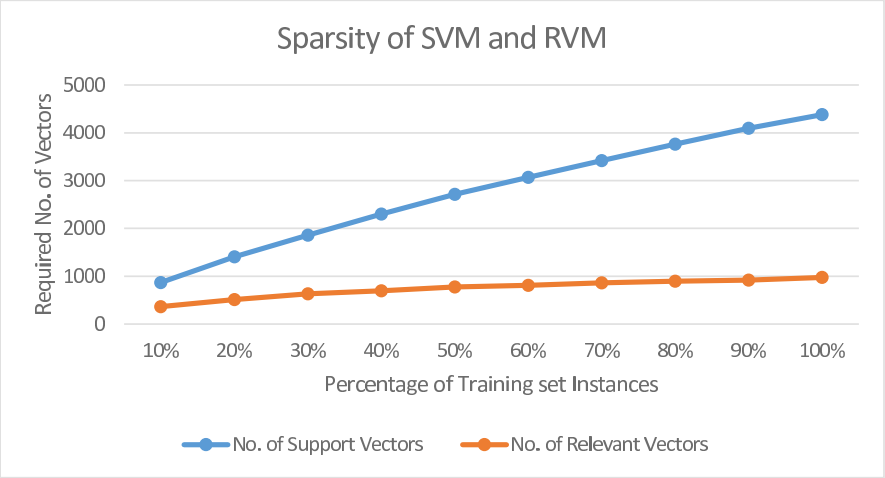

Figure 4.3: Comparison of the Sparsity Ability of SVM and RVM

sample size, the number of required support vectors for SVM classification grows faster than the number of required relevance vectors for RVM classification. In other words, the RV/SV grows linearly with the growth in training instances.

4.5.2 Performance Comparison in Terms of Generalization

Generalization ability is an important factor in machine learning techniques. To compare the generalization ability of both classifiers, we derived classification error rate, which can be easily computed based on accuracy of the classification algorithm. Since the increase in the error rate in RVM is less than the increase in the error rate in SVM with respect to increase the testing set, RVM has better generalization ability than that of SVM. The performance comparison of two classifiers, RVM and SVM are depicted in Fig. 4.4.

4.5.3 Performance Comparison in Terms of Classification

The performance comparison between RVM and SVM is evaluated using accuracy, precision as well as the area under the ROC curve and F-measurement. Table 4.3 demonstrate that SVM

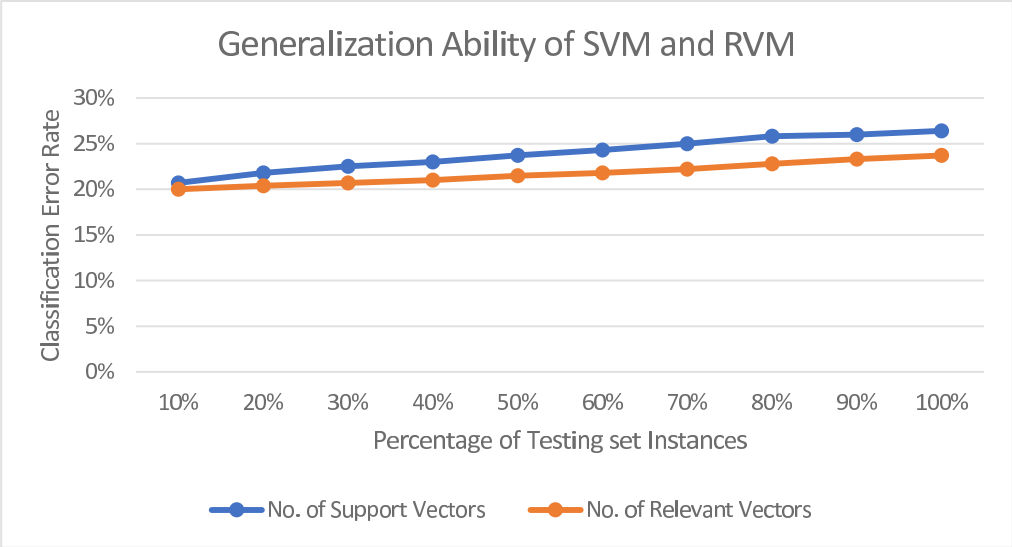

Figure 4.4: Comparison of the Generalization Ability of SVM and RVM

Table 4.3: Classification Comparison between SVM and RVM

Kernel Function	Accuracy	Preciasion	F-measure	ROC
SVM	72.3	81.5	**71.5**	75.4
RVM	74.2	83.3	73.4	77.3

and RVM have comparable classification results. However, RVM classification is slightly better than SVM.

4.5.4 Performance Comparison in Terms of Complexity

SVM and RVM both have complex mathematics and optimization problems to solve in order to build the training model. The complexity of these classification techniques depends not only to the training model construction, but it also highly depends on computation process that the classifier is involved with. SVM complexity is in order two, $O(n^2)$ while RVM complexity is in order three, $O(n^3)$. Therefore, RVM is more complex that SVM. Therefore, SVM are much faster than RVM even though fewer number of data point are required and only because RVM

required high-level computation which requires more time to train the model. However, on the testing, on the other hand, RVM is much faster than SVM.

4.6 Summary

A comprehensive study on the effect of different kernel techniques on well-known support vector machine is presented. Moreover, relevance vector machine is used to classify the network traffic instance. The advantages of using RVM over SVM relies on the fact that, first, RVM requires less relevance vector machines comparing to the required support vector machine in SVM. Second, RVM provide a probabilistic interpretation of the prediction. It basically means than unlike almost all the classifiers that provide black and white decision on whether a traffic instance is either normal or anomaly, RVM provides prediction decision based on the probability of prediction on the behavior of that network traffic behavior instance. Based on the comparison between classification algorithms, SVM with linear kernel has better performance in terms of accuracy, precision and detection rate; however, on the other hand RVM with much less number of relevance vectors has slightly less detection rate comparatively. Based on the simulations results, RVM is sparser than SVM; i.e. the number of relevance vector increases slower than that of support vectors with the grow in size of the training set. Moreover, RVM is much faster in making decision than SVM and have better generalization ability than SVM. However, the performance of SVM and RVM are comparable. The main advantage of relevance vector machine over support vector machine is that, despite of the SVM which provides a hard prediction, black and white decision about the behavior of an instance, RVM find the probabilistic prediction of the class label.

Chapter 5

Cross-layer Trust-based IDS for Malicious Packet Loss

5.1 Introduction

Mobile Ad hoc Network (MANET) is a multi hop wireless network consisting of self-configurable mobile devices distributed in a geographical area with dynamic topology due to mobility nature of these networks. MANETs are equipped with wireless transmitter and receiver with bi-directional links that communicate either directly or indirectly with other nodes in the network. Direct communication represents one-hop communication in which source and destination are within the same communication range. However, in the indirect communication, source and destination have different communication range and therefore, the network relies on the relay nodes and multi-hop communication to transmit information. In other words, every node is both transmitter and receiver because in cases where the destination is not in the same communication range with the source, the intermediate nodes should relay the message

to the destination [**?**]. In such networks, source node relies on the neighbor nodes to relay data through multi-hop communication.

This type of network has become a necessary facet of our lives and the mobility nature of these networks makes it a very interesting wireless network for the situations where the nodes are required to have movement or in cases where providing and managing the network infrastructure is costly and not efficient. MANETS have various applications such as military activities, disaster discovery, medical emergency discovery grows every day. In addition, these wireless mobile networks can be applied in management and commercial services. For instance, MANET can be used to establish a communication system for taxi cabs to provide passengers information about the route directions, weather condition etc [25].

Despite all the potentials and interesting characteristics of MANETs, like every other wireless communication networks are not immune from intrusive attacks and are even more prone due to inherent characteristics of these networks such as its mobility nature, since nodes can easily join and leave the network, lack of infrastructure and lack of a centralized controlling unit and administration as well as an open medium of MANETs, results in having attackers come from all directions since there is no boundary for the network. Intrusive attacks can be very harmful sometimes and have severe effects on the whole network performance and therefore the need to identify and prevent it to further damage the network is essential. From an application point of view, a robust MANET should be resilient to Denial of Service(DoS) attacks and have the capability to identify attackers and prevent further damages because insecure environment and fundamental characteristics of these networks make it very challenging to be applied in various applications [90].

Among all layers of the protocol stack, the network layer is more prone to different kinds

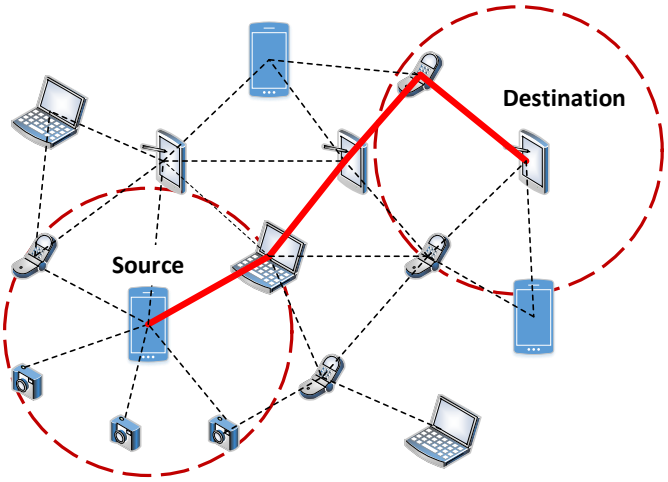

Figure 5.1: MANET Topology

of attacks due to use of cooperative routing algorithms, limited battery resources and computational ability as well as the transient nature of the wireless communication [25]. One of the most destructive malicious activities of an intruder is packet dropping. The malicious node drops the received packet deliberately with the intention of causing disruption in the routing. Since packet dropping is not always caused by an intrusive node, meaning that a node might drop the packet due to low battery, congested buffer, etc. Since packet dropping can have several reasons, in order to find the malicious node, first, it is essential to know that the packet dropping is caused by a malicious node only [82]. In order detect the packet dropping caused by a malicious activity of an attacker, we need to use information from various layers of OSI protocol to confirm whether it is caused by a compromised node deliberately dropping packets or due to other non-malicious reasons.

Many studies have been done on security for wireless mobile networks, however, only a few have pointed out the prospect of intrusion detections. Some general approaches to secure

routing and integrity of such networks are a key generation, authentication, cryptography and many other proposed schemes. However, the major problem with these approaches is that they all have their own drawbacks and can sometimes be relatively expensive and not applicable for many purposes. Despite various security mechanisms such as cryptography, message integrity, coding, authentication, key management and many others that are proposed to avoid security threats, wireless networks still suffer from many security vulnerabilities such as Denial of Services(DoS) attacks and these security mechanisms are not sufficient to identify various types of attacks. The major shortcoming is to identify the internal attacker efficiently. low-overhead preventive security methods have been proposed for wireless networks, such as key exchange, ciphering algorithms, hashing methods to guarantee the confidentiality, availability, and integrity of these networks. However, they are incapable of protecting the network against a disruptive attack that threatens the traffic flow in the network [45].

As discussed earlier in this dissertation, intrusion detection system is a promising solution for network attack detection which can be applied to MANETs as well. However, the mobility of the nodes, as well as the wireless link for communication, raises the challenge for IDS to effectively detect attacks[1]. Moreover, since the objective is to detect and remove malicious nodes that drop the packets deliberately, it is important to understand other reasons that can cause packet loss and be able to distinguish whether the packet loss is caused by malicious nodes or is due to faulty networks. Since, single layer IDS have limited access to the network features (i.e. have access to the data provided by the protocol layer where the IDS resides), it cannot efficiently detect malicious packet loss and consider other faulty network causes of network packet loss. In this regard, conventional single layer IDS cannot detect malicious

[1]Since the network in MANETs is decentralized, the IDS resides in each and every node of the network [44]

packet loss effectively.

Therefore, the problem of malicious packet loss detection in MANET considering features from multiple layer of the protocol to eliminate the uncertainty about packet loss due to faulty network as hence increase the accuracy of detection is studied in details in this chapter. In order to enhance the probability of detection and reduce the false positive rate, a novel cross-layer protocol design which enables information exchange across layers possible is proposed. Moreover, trust based mechanisms is used to distinguish between malicious packet losses and other causes of packet losses. In this study, we have considered the cases where packet dropping could be caused by low energy, congestion and other faulty network reasons. In cases where the node has low energy, it refuses to relay and drops all the received packets since it does not have the sufficient energy to transmit the packet or when two or more packets are received by a node simultaneously, it has no choice but to drop one packet since its buffer space is limited and incapable of accommodating more than one packet. Similarly, a non-legitimate node attempting to access the shared medium at the same time can cause a collision which results in packet loss.

To this end, a novel cross-layer design for detection of malicious packet drop in MANET is proposed in this chapter which uses three layers of the protocol namely: the physical layer, network layer and MAC layer that collaborate to provide a better detection decision. Trust-based mechanism is used in this study for malicious node detection. The main contribution of this work relies on addressing the problem of high false alarm rate in single layer intrusion detection. In this study, we take advantage of using cross-layer design to have a better detection of the malicious node dropping packets and excluding the node from routing path in MANETs.

5.1.1 Cross-layer Intrusion Detection Technique

Here we discuss a number of intrusion detection systems have been proposed in the literature. In [91], the authors proposed a cross-layer design where every layer has its own individual detection module and the correlation of the output from every layer obtain the detection decision. A novel reliable routing using cross layer metrics is proposed in [64]. The combination of traffic features from both MAC layer and network layer to provide more reliable routing path in the network layer.

The authors in [82] have considered features from network layer and used them as metric to distinguish malicious packet loss from packet drop caused by mobility. In this study, after the detection process, link layer is responsible for updating the list of the neighbors and neglecting the malicious nodes. The study in [18], proposed a cross-layer design that correlates the detection results from both single layer sessions as well as sessions taking place in multiple layers of the protocol based on fuzzy logic mechanism.

The author considers using MAC layer and network layer to identify an intrusive attack in the wireless network using AODV protocol in [51]. Principal components of the training set are used to find the important components which reflect the distribution of the training data set. then based on the value of the projection distance for each node as a threshold, the normality of nodes is presented.

In [3], the authors proposed a cross-layer based on detection of new patterns of the routing traffic, prioritizing them and adaptation to incremental updates of the detection model. In data collection part, the proposed method collect information from across layer of the protocol from traffic to dynamic topology and routing behavior. Next, the collected data are monitored, and

new classified patterns are classified followed by data reduction and learning module to validate the decision.

A cross-layer design to efficiently detect intrusions in wireless networks is proposed in [80]. In this work, the module to the link between layers and IDS is also proposed to reduce overhead during the data collection. A clustering algorithm has been used for anomaly detection. Anomaly cross layer intrusion detection system has been proposed in [98] in order to increase detection rate and reduce false alarm rate, using the interaction among physical layer, network, and MAC layer in wireless mesh networks based on machine learning algorithms.

In [13], the authors considered only the collision as a legitimate reason for packet loss and other reasons has not been taken into account. The major difference of proposed approach with [76] is that it did not consider legitimate causes that can result in packet loss in the network. Despite all of the above-mentioned cross-layer designs, a cross-layer IDS is proposed in [73] where the detection mechanism is shared among different layers of protocol which can be interpreted as a system that has IDS in each layer of its protocol design. The author believes that due to the presence of different kinds of protocols across each layer, it is difficult and at the same time not effective and efficient to use a single layer detection mechanism. Moreover, the author proposed cross-layer with the same approach of having IDSs at each layer in [90]. However, it has been proven that considering individual IDSs at each layer where the layers will exchange the detection information. can result in significant processing over the head. Thw study in [92] discussed the two possible cross-layer interactions: either through a structured shared database where every layer interacts with a common shared database, which an interface between different layers of the protocol and detection component, or through information exchange using direct interactions between two adjacent or on adjacent layers of the protocol

and their advantages and drawbacks. In the latter case, when a malicious activity in one layer is detected it triggers the detection engine of other layers to confirm the attack in the network. The authors suggested that if packet dropping attack is detected using watchdog mechanism, in order to confirm the existence of this attack only due to malicious behavior of the network, confirmation from other layers are required. Although this is a valid logic, however, since, in this type of interactions, only two layers can exchange information at a given time slot, the efficiency of the detection system degrades due to the delay in interactions of the layers. Other potential shortcomings of direct interactions between layers are:

1. Not every direct interaction is helpful in performance enhancement.

2. More collected and shared information results in higher overhead (There is a trade-off between false positive rate and overhead cost).

3. Loops of detection information exchange between layers has an adversary effect on stability.

4. The no-independent functionality of different layers of the protocol stack would lead to loss of modularity.

In this cross-layer design, the modularity of the network is preserved due to the fact that only necessary information is exposed to other layers through an interface. Moreover, since layers only interact with a shared data based and it is responsible for coordination of information and detection, the chance of the loop creation would be minimized and therefore the would be more stability in the network.

Due to the above-mentioned disadvantages of the direct interaction cross layer design, im-

plementation complexity, and based on the ease of management of the first approach, author suggests that using a shared data based is preferred rather than using the direct interaction between layers in [92].

Among all the studies in this research area, there are four well-known approaches for packet loss detection, namely: watchdog, TWOACK [14], AACK [7] and EAACK [50]. Watchdog is an efficient IDS that was proposed for MANET and many studies used it as the basis of their work, however, watchdog detection mechanism fails to find malicious misbehaviors in the presence of a collision, limited transmission power, collusion, partially dropping [**?**]. TWOACK is presented to tackle with the shortcomings of the watchdog mechanism and it is based on the acknowledge message received from the next two hop to the source. It addresses the problem of collision and limited transmission power; however, unlike watchdog mechanism that reports malicious nodes, TWOACK can only identify misbehaving links. Moreover, this method suffers from a significant amount of overhead. Similarly, AACK method relies on acknowledgment from other nodes participating in the transmission. It has less amount of network overhead comparing to TWOACK with similar throughput. The main issue with the following IDSs is that they are still very prone to the false misbehavior report. In cases a node transmits ACK packet to the source even though it has not received any packet, the system fails to identify malicious behavior in the network. When using acknowledgment-based detection approach, it is crucial to validate the ACK packet and authentication; otherwise, the system would be unable to identify intrusions. In another word, ACK packet is not a very reliable measurement metric for IDSs. To address this issue, EAACK is proposed in [**?**] which adopt digital signatures. Authors in [30], proposed a novel intrusion detection to detect packet dropping in MANET. The malicious node is identified using the comparison of the information of

the normal routing path information and the path that is chosen by the attacker.

In [49], the author suggested there are two types of cross-layer IDS based on the number of detection analysis units. The first category collects data from different layers and also has multiple data analyzing units in every layer of the protocol. The detection decision is made by the weighted correlation of the prediction from IDS in each layer. The major drawback of this design is that having IDS at each layer significantly increase the overhead. However, in the second category, all the collected data are used in a single detection analyzer which is located in one optimal layer. In other words, the detection units perform independently of other layers. The behavioral information and s from different layers are collected and all the processing for malicious behavior detection are done in a single layer. In fact, the correlation among features is the main essence of this cross-layer IDS design. Our work is based on the latter case in which we consider the weighted correlation of features from 3 layers in order to find malicious behavior in routing which is in the network layer.

Cross-layer design is one way of using the shared information among layers [43]. Cross-layer IDS provide a collaborative decision based on information from different layers. This type of collaboration has the potential to offer higher detection accuracy and a low number of false positive rate. Many IDSs have been proposed in the literature to identify the intruder node however they mostly operate in a single layer of the OSI model or network model and do not consider the collaboration and interaction among different layers. However, in cross-layer design, the information is exchanged among different layers and trigger multiple levels of detection and use the knowledge of network [60]. Cross-layer design focuses on analyzing and sharing statistics from different layers. In cross-layer design, the selection of the right layers of the protocol is very critical to efficient and effective anomaly detection. Cross-layer approach

monitors the network, communicate with different layers and exchange information. One of the major concern with conventional implemented IDS is that they are misused-based IDS and due to the rapid change in the appearance and effects of attackers, and considering that misused-based detection suffers from low efficiency in detection of novel attacks, the predefined set of attacker's signature requires to be updated more frequently than before which is not suitable for accurate attack detection. In addition, current IDS only consider one layer of the protocol to detect compromised node in the network. However, some attackers can affect multiple layers of the protocol and it differs from one layer to another. Hence, it is essential to consider multiple layers into consideration for anomaly detection in the network traffic data [80].

Cross-layer IDS is an alternative solution to overcome the drawbacks of conventional IDS and they have higher detection accuracy since more information is involved in detection of the compromised node [12]. The wireless medium has caused the network to be more exposed to different types of attacks. On the other hand, independent security mitigation techniques in different layers may result in a conflict of interest and would lead to performance degradation [92]. Robustness in cross-layer IDS design is a very important factor. The need for cross-layer IDSs is inevitable due to several factors.

1. Attacks have become more sophisticated nowadays and conventional IDSs are not sufficient to detect such attacks and multiple levels of observation or maybe detection is required to first, find the essence of attacks and second, to improve the detection accuracy at the same time.

2. The performance of single layer IDS is not sufficient and cross-layer IDS has relatively better performance in terms of accuracy and false positive rate

Cross-layer IDS utilize the shared information about the traffic patterns and attack traces from nodes within the same communication range from multiple protocol layers. The ultimate goal behind cross-layer approach for intrusion detection system is as follows:

- To detect multi layer attacks using multilayer features

- To enhance detection performance using parameters across layers of protocol stack.

In the proposed cross-layer design, different layers of the protocol, namely: network layer, MAC layer, and physical layer collaborate and share information using a shared database to exploit the information available across different layers and make it available to different on adjacent layers. Collaboration among network layer, MAC layer, and the physical layer can significantly enhance the detection of the intentional malicious packet loss in the network by using cross-layer design. The objective of proposed cross-layer intrusion detection is to detect malicious packet loss more accurately, while obtaining a more reliable and secure route from source to the destination through collaboration among different layers of the protocol via shared information database to improve efficiency and the performance of MANETs. Moreover, in this study, the best routing path for data transmission is selected excluding the malicious node.

Cross-layer IDS are generally categorized into two information sharing mechanism. In centralized cross-layer IDS, the detection takes place in only one layer and other layers only share and provide the required information and features to the layer that is responsible for detecting the layer-specific attack; however, unlike the centralized mechanism, in the distributed cross-layer design, every layer is responsible for observing the network and identifying any malicious activity and share detection results to other layers. In other words, in the distributed cross-layer design the detection mechanism and the suspicious malicious nodes are shared among different layers of the protocol and with this collaboration the protocol decides whether the suspicious

node is an attacker or not. The collaboration among different layers gets confirmation from different layers that the suspicious malicious node according to one layer is a malicious node or not [60, 91].

Multiple Data Collection Multiple Data Analysis

In Multiple Data Collection and Multiple Data Analysis (MCMA), as [49] suggested, there IDSs are in one layer of the protocol stack and the confirmation from each layer is required to make a decision about the suspicious node. It can be interpreted as each layer has its own detection mechanism and the correlation among detection decision of layers will provide the final decision about the existence of an attacker in the network. The proposed cooperative cross-layer detection in [13] is based on this technique. In this approach, first, the information about the malicious behavior is obtained from different layers of the protocol stack and secondly, the identification of the malicious node and a final decision is based on the weighted and correlation of the prediction from individual layers. This indicates that each layer is equipped with its own IDS [90].

Multiple Data Collection Single Data Analysis

In Multiple Data Collection and Single Data Analysis(MCSA), the features across layers are collected and used in one single layer which equipped with IDS. Using this technique, only one detection analyzer is available for each node and may reside in any of the layers of the protocol depending on the attacker type; however, it is notable that the features and data are collected not only from the layer where the detection system resides but also from other adjacent and non-adjacent layers. Despite MCMA approach, in this cross-layer scheme, the detection unit resides

in one layer using the obtained information from itself and other layers. The main advantage of this scheme is that the energy consumption decreases significantly [90]. Collecting information from various layers helps to confirm the malicious behavior in the network and detect malicious nodes with more accuracy [16]

5.1.2 Trust-based Compromised Node Detection

A secure trust based intrusion detection system considering both direct and indirect trust value calculation is proposed in [72]. Similar to EAACK approach, this technique is also based on the acknowledgment packet. After receiving the RREQ message and checking the destination IP, it sends ACK message to the source to represent that it has a path to the destination and if the ACK message is received successfully the trust value of the path will increase, otherwise, the source would look for a malicious attack. Authors in [38] proposed a trust-based ADOV protocol design; however, the calculation of the trust value is not clearly stated. One possible way to ensure secure communication is to have all communicating nodes trusted by building a trust model [46] It is suggested that trustworthiness of the node in the network should not be limited to the communication point of view and other metrics should also be taken into considerations. In [21], the author considers the case that there are different intrusion detection in a different layer and in order to determine a node to be malicious all the IDS existing in different layers need to confirm it. There is a no trust region where suspicious nodes with packet dropping, collision and misdirection attacks are put into and the detection continues. In [76] the author proposed a cross-layer design in MANETs that utilize the physical layer features such as transmission range, link stability and node degree to define the trust value of the nodes

in the network. The node with the highest trust value will be chosen as an observer node and is responsible for identifying the malicious node by considering the number of received and forwarded packets. If a node only intercepts packet and do not forward them, the node will be considered as a malicious node and its trust value will decrease. Also using the proposed trust value, the route with the highest absolute value is considered to be the path for packet transmission and MAC layer allocates the access time and access control. At the end, the node with the least trust value is detected as an attacker. Despite what the author claimed, the node with the least trust value is not necessarily an attacker due to the fact that we need to compare the trust value of nodes with their initial trust values and consider the deviation of trust value from its initial point.

5.2 Attack Types and Categories

In general, attacks can be categorized in three different ways. First, they are either active or passive. Second, they are either internal attacks or external attacks. Third, based on their target layer of the protocol.

5.2.1 Active Attacks Versus Passive Attacks

Security attacks against wireless ad-hoc networks are categorized into passive and active attacks. Passive attacks eavesdrop in the network to collect information. However, on the other hand, active attacks cause trouble in the network and the attacker disrupts network operation. Active attackers cause disruption in the network operations; however; the passive attackers only monitor the network and collect information about the network to provide them to the ac-

Table 5.1: DoS attacks in different layers of the protocol

Protocol OSI Layers	Attack
Physical	Jamming, Tampering, eavesdropping, scrambling
Data link	Collision, Exhaustion, Unfairness, Disruption MAC
Network	Neglect and Greed, Homing, Misdirection, Blackhole Greyhole, Packet Drop Sleep Deprivation, Sybil Sinkhole, Hello Flood Selective Forwarding Spoofing and altering route information
Transport	Flooding, Desynchronization Session Hijacking

tive attackers. Spoofing, packet modification, black hole and grey hole and wormhole attacks, as well as all DoS attacks [24] are examples of active attacks while eavesdropping and traffic monitoring are considered as passive attacks [8]. Passive attacks do not disrupt the network and only monitors the network to collect the valuable information by listening to the network traffic. Passive attacks can obtain information about the network topology, the location of the nodes and their identity and provide this information to the active ones. On the other hand, active attacks try to gain unauthorized access and degrade the performance of the network [4].

5.2.2 Internal Attacks Versus External Attacks

Active attacks can further be categorized into two types of internal and external attacks. External attacks that are carried out by a node that does not belong to the network and internal attacks are the ones that a node from inside the network acts maliciously. Internal attacks are more severe and difficult to detect. Most of the studies in the literature address the external

Table 5.2: Parameters in different layers of the protocol

Protocol OSI Layers	**Parameters**
Physical	Channel switching frequency Energy Level
Data link	Bandwidth, link loss rate Access delay No. of medium access Duration of medium access
Network	Data packet, route request, route reply and route error, link added, link modified, route changed

attack detection; however, the efficient detection of internal attacks remains a problem which this study addresses it. Insider or internal attacks lie within the MANET and can cause severe harm to the network. On the other hand, the external attackers are not a member of the MANET. In comparison to the internal attackers, they are considered to have less harm to the network [8]. The internal attackers can be the legitimate nodes with shared key information and might participate in the routing process. These malicious attackers can disrupt the routing process and can attack routing discovery phase or route maintenance phase [8]. Internal attacks are harder to be mitigated since they have all the information about the network.

5.2.3 Layer-specific Attacks

Attacks can be further classified based on the layer they aim to cause disruption in. A list of attacks with their target layer is presented in table 5.2. For instance, jamming and eavesdropping attacks disrupt physical layer while collision and packet drop attacks threaten data link layer and network layer, respectively.

5.3 Proposed Cross-layer Trust-based Malicious Packet Loss Detection Architecture

As discussed earlier, the conventional methods failed to determining the real cause of packet loss and the proposed protocol designs that aims to detect malicious packet drops ignore legitimate or natural causes of packet losses. These shortcomings motivated us to propose a technique that can distinguish malicious packet loss from other packet loss causes. In this study, packet loss refers to any kind of attacks that causes destination node to not receive transmitted packets from the source node and non-malicious packet loss refers to all the legitimate reasons due to faulty networks. Packet drop is not always due to the existence of the malicious node. This misbehavior of the network can be due to congestion, lack of energy, poor channel condition, for instance. Here, a brief description of each scenario is represented. If the intrusion detection system is not well designed, the detection of malicious packet loss will be with high false positive rate due to the above-mentioned legitimate packet loss reasons. Figure 5.2 demonstrates the three main components of the proposed cross-layer intrusion detection design.

5.3.1 Data Collection and Feature Extraction

This unit is responsible to observe and monitor the events, preprocessing the captured data from the network, extract features from physical, network and MAC layer and select the informative features about packet loss detection, share all collected features in a shared data based for all the contributing layers for access and finally present them to the intrusion detection unit.

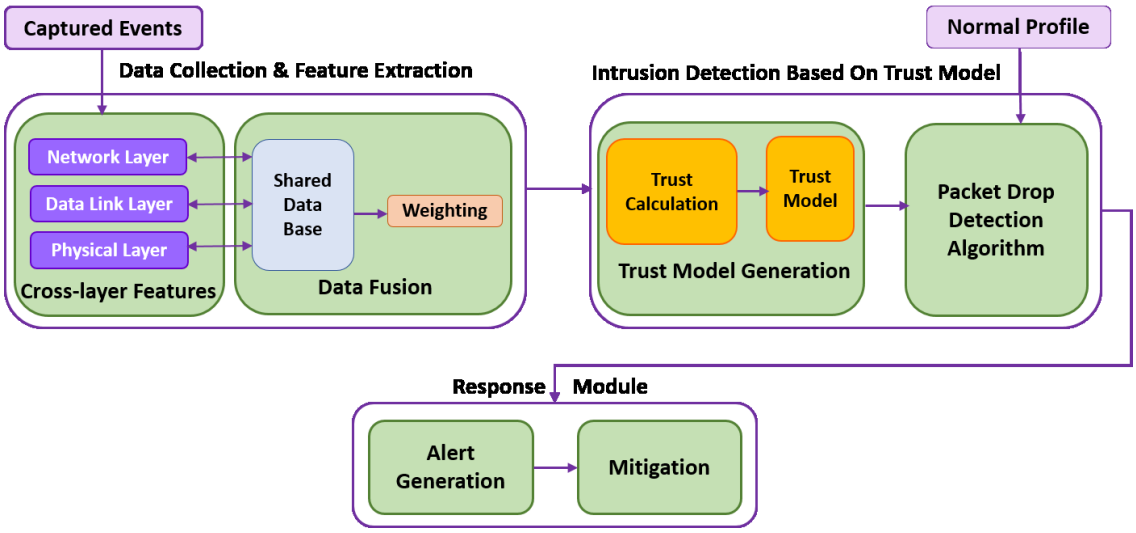

Figure 5.2: Proposed Cross-layer IDS Architecture

Cross-layer design helps to identify the correlation and relation between features from various layers [51].

Cross-layer Features

The right combination of layers is crucial in the design of a cross-layer IDSs. Moreover, it the right selection of layer statics is extremely important in cross-layer design. For instance, a collaboration of network, MAC and physical layer are the best for routing attack detection and to identify DoS attackers, it is better to include MAC layer features and statistics [49]. The parameters or in other words, the information that is used and shared in the shared database is extremely important in the proposed cross-layer protocol design since the lack of sufficient information significantly degrades the performance of the network. Network traffic statistics collected from both MAC layer and Network layer are essential in proposed technique. Moreover, physical layer characteristics also play an important role in detection mechanism. In network layer, the forwarding behavior of the neighbor nodes is considered while in MAC

layer, the wireless link quality is considered. The network layer is responsible for establishing an end-to-end connection over the network and is responsible for discovering the route for the packet to reach the destination.

Transport Layer Features all the number of RREQ messages, as well as RRER and RREP messages, are available at the transport layer.

Network Layer Features All routing information are taken from network layer.

Link Layer Features MAC layer has information about the congestion and interception [44]. It is responsible for providing a secure and reliable link in the wireless network. This layer allocates time-frequency or code spacing among mobile sharing wireless channels [43]. Information available in this layer are but not limited to the number of active data transmission, the delay between data transmission rate and the number of data retransmissions. Moreover, the link layer is responsible for keeping track of the mobility of nodes and updating the list of neighboring nodes at every time instance. In the proposed technique, after defining the secure routing path based on the trust model, the source node provides the secure route information to the MAC layer to allocate less access time to the nodes with less trust value and give more access time with nodes with higher trust values. In other words, MAC layer ensures that the packet loss is not due to collision and therefore is due to buffering congestion. It counts the number of sent RTS packets and received CTS packets. Any difference indicates that the packet loss is due to the collision.

Physical Layer Features Transmission power and bit error rate are the major contribution of physical layer in malicious packet loss detection.

5.3.2 Intrusion Detection Based on Trust Model

When a packet is sent from node A to node B, after receiving the packet, node B can either relay it to the next hop that is either destination node or another node in the routing path to the destination, or it can be intercepted and drop the packet. There are several reasons that can cause this failure to node B to receive or relay the received packet to the next hop. If the buffer space of node B is full, it is incapable of accepting new packets and therefore drops the packets due to buffer overflow. Channel error, on the other hand, can corrupt the packet content and force node B to drop it since it is not readable [42]. In order to find the packet drop attack due to malicious activity of an attacker node, we need to identify other motivations of a legitimate node to drop the received packet. Through literature, this issue has been widely investigated and possible motivations are described as: collision, the constraint is energy resources or even mobility. A collision occurs when nodes try to access the shared medium at the same time. Another legitimate reason for packet drop is the corruption of the packet due to signal losses, interference or a high bit error rate.

In order to confirm that the suspicious malicious node which drops packets is an intrusive node or in other words, a node drops the packets deliberately, each node in the network should calculate the trust value of all its neighbor nodes based on its link stability, residual energy, congestion and collision which are the legitimate causes of packet loss in the network.

In the beginning, it is considered that every node has a trust value of 1 in the network.

With each data transmitted in the network, the trust values of each node are updated. This trust value is then used in route request message which is further secured with the help of message authentication code(MAC) calculated for each node.

The trust value of the destination is updated once it receives the route request and this value will be sent to the source via RREP message. Then the source selects the route with the highest trust value. Finally, the path information is presented to MAC layer for access time and access control allocation which is also based on trust values.

Trust Model Generation

Trust is a concept that determines the availability, reliability, and quality of the service of a node. Trust can be defined as the belief one node has about other nodes in the network based on their behavior or recommendation from other nodes. The authors in [61] referred trust to the belief one node holds on another node about the ability to successfully forward the packets. Trust refers to the belief that one node holds about another node based on previous knowledge or recommendations from other nodes [61]. Continuous evaluation of the performance of nodes is vital to learn about the network behavior and it is considered for calculation of the trust value of a particular node. However, in this study, we refer to the trust as a measurement of the capability and ability of a node to forward the received packets. Using trust models to meditate the network against various kinds of threats has become an essential aspect of MANETs.

Typically, there are two types of trust model establishment. It can be either based on calculation and recommendation from a third party which is an indirect way to obtain the trust value of a node or it can be based on direct interactions [46]. In direct observation, an observer estimates the trust value of one hop neighbors. Researchers use other nodes opinion or indirect

trust observation to not be biased to the decision of one observer [99]. In direct observation the monitoring node is the same node as the one evaluating the trust value; however, the indirect observation, the other nodes, third party, collects the required evidence for the node that is evaluating another nodes trust value. Indirect trust is the estimation of other nodes about the trustworthiness of a target node that is collected by an observer node.In other words, the legitimacy of the neighbor node that provides feedback on the trustworthiness of a specific node is not clear to the collector node. A malicious node can deliberately provide false feedback about the target node. Hence, the uncertainty of the indirect observation information is an important issue and many works have focused on trust establishment based on unreliable indirect observation [99]. Hence, since the trustworthy of the third parties is an issue itself and there is no guarantee that the third parties are non-malicious nodes and also getting feedback from neighbor nodes would increase the communication overhead which is not desirable investigation on the reliability of the indirect trust observation is not in the scope of this study and we considered the direct trust observation in our work.

In this study, a trust based AODV protocol has been proposed for MANET that its measurements are from different layers of the protocol. This Trust-based Cross-layer AODV protocol (CTAODV)[2]. In this study, the trust value range is between 0 and 1 where 0 means no trust while 1 refers to certain trustworthy of a node. Moreover, if the trust value of a node relies on the range of [0, Δa] it is considered as a malicious node and if its trust value falls in the range of [Δa, 1], it is considered to be trusted and a normal node in the network. Where Δa refers to the predefined threshold[3] for trust degree calculation. It is notable that this threshold can be

[2]It worth mentioning that in this study we adopt MCSA where we collect data from the network, link and physical layer and the detection or analysis unit resides in network layer to identify packet losses.

[3]It is notable that all the threshold values used in this study and proposed algorithm are defined by historical data. For instance, based on previous observation of the network behavior, it is known that if the residual energy

easily defined based on the previous observation of the network. Statistical threshold values for each parameter or features extracted from various layers of the protocol can be easily calculated during the training phase. The trust value that one node can hold from another node in its neighborhood can be defined as below where T_x^y represent the trust value that node x holds from node y as shown in equation 5.

$$T_x^y = \sum_{i=1}^{k} [w_i(A).T] \quad where \quad 0 < T_x^y < 1 \tag{5.1}$$

$$w_1 + w_2 + w_3 + w_4 = 1 \quad and \quad 0 < w_1, w_2, w_3, w_4 < 1 \tag{5.2}$$

Equation 5.1 shows that the overall trust value that one nodes from another node is the summation of the weighted individual trust models. In this study, we define individual trust models as the legitimate reasons for packet loss. The summation of the weights of individual trust models should not exceed one.

Collision:

In the proposed collision detection scheme, we introduced a novel scheme to enable the source node to identify the collision in the network. When the transmitter does not receive acknowledge after the transmission starts, it selects the node with the highest trust value in the communication range of the transmitter and the relay node and assigns it to monitor the network. This area is the best choice for the monitoring node to reside since the malicious node has to be in the relay node transmission range in the worst case scenario in order to induce collision. This monitoring node should be able to hear the from both the transmitter and malicious node intending to cause a collision. It is responsible to observe the network and detecting the nodes

of a node drops below a certain value the node is not capable of participation in the transmission.

that transmit at the same time and have one similar destination. It then provides a list of suspicious nodes to the transmitter and now it's the responsibility of the transmitter node to identify the malicious nodes based on the repetition of a node(s) in the suspicious list provided by the monitoring node. Since mobility is one of the undeniable parts of the MANET, it is important for the transmitter node to update the monitoring node periodically since one selected monitoring node may not be in the preferred communication range after some time. In this case, the transmitter should assign a new monitoring node and get the suspicious list from the newly assigned monitoring node. The probability of a collision-free and error-free communication can be calculated based on 5.3 and 5.4; respectively. Therefore, the probability of having packets receive the intended destination without collision and error in the link can be calculated based on 5.5.

$$P_{collision\ free} = (1 - P_{col}).(1 - P_{col}(1 - P_{col})) \tag{5.3}$$

$$P_{error\ free} = (1 - P_{err}) \tag{5.4}$$

$$T_{correctly\ recieved} = P_{correctly\ recieved} = P_{collision\ free}.P_{error\ free} \tag{5.5}$$

Since P_{col} is related to the traffic rate, it should be calculated based on the number of RTS and CTS and in the case where these two packets are not received within the specific time period, the probability of packet loss due to collision will increase. The probability of collision occur in the network between two nodes can be calculated using the following formula:

$$P_{col} = \frac{(\#RTS - \#CTS)}{\#RTS} \tag{5.6}$$

Link Stability:

Received Signal Strength(RSS) alone, cannot be used to determine the links status due to the mobility nature of MANET. Mobility can cause the network to be unstable and therefore RSS is not a good indicator for link status. Alternatively, link stability can be used to measure the connectivity between two nodes in the network. Link stability is defined by the transmission range and distance between two nodes as follows:

$$T_{ls} = Link\ Stability = \frac{Transmission\ Range}{Distance} \tag{5.7}$$

In general, the transmission range can be derived using the following equations:

$$P_r = 10\log P_t + G_t + G_r + (20\log\frac{\lambda}{4\pi R}) + 10\eta\log(\frac{R}{D}) + P_{shadow\ path} + P_{fading} \tag{5.8}$$

Where P_r is the received power and P_t is the transmission power into the antenna while G_t and G_r are the antenna gain of the transmitter and antenna gain of the receiver respectively. λ is the wavelength of the signal and R is the distance between the transmitter and receiver antenna and D is the transmission range. Hence, in order to derive the transmission range, knowing all other parameters in equations 5.8, D can be calculated. Moreover, $P_{shadow\ path}$ and P_{fading} are the power due to shadowing and power due to fading, respectively. η is the path loss component. It is notable that all the gains are in dB and therefore, the transmission range would be in meter. Communication range is used since the network should know whether the abnormal packet loss behavior is due to having the relay node moving and getting out of the communication range of the transmitter node or the packet loss is due to the malicious activity

of the relay node.

Residual Energy:

In every data transmission, the nodes that are involved in transmission lose energy at every data transmission. At a critical energy level, the node is not able to participate in packet transmission due to its low energy. In this case, packet loss is not due to malicious behavior. Instead, the network and nodes are working normally and the packet drop is only because of the lack of resources. As the authors suggested in [76], the residual energy is the remainder of the node's energy after a certain amount of time being active in transmission. It can be computed based on the subtraction of the consumed energy while knowing the initial energy level of each node. This is one of the very basic physical characteristics of nodes in wireless networks. The probability of the residual energy can be easily derived based on the above-mentioned equations

$$T_{re} = \frac{E_I - E_C}{E_I} \tag{5.9}$$

Equation 5.9, shows the calculation of the trust model based on the residual energy where E_I is the initial energy level and E_C is the consumed energy level of the same node that participate in data transmission. The energy consumption level can be computed based on the following:

$$E_C = \rho N_F N_R E_b + \rho N_r E_b \tag{5.10}$$

where N_F and N_R are the number of forwarded packets and received packets while E_b is the energy level per bit and ρ is the packet size in bits.

Congestion:

Congestion can be measured and estimated using the free buffer space in the link layer. In order to have congestion-free network and data transmission, only the nodes with sufficient buffer space should forward the RREQ packets and participate in the routing. If the buffer level is pretty low and cannot accommodate any more packets, it will drop RREQ packets. Therefore, by comparing the number of RREQ and RREP, one could have a good estimation of the probability of having congestion in the network. Congestion occurs when a node tries to carry more data that it can accommodate is capable of handling [20] or when multiple users try to access the available resources to transmit their information and data. Congestion is one of the main reasons for packet loss in the network.

$$Ret(x) = B_c(x) - \sum_{i=1}^{N} m_{size}^{i}(x) \tag{5.11}$$

Retentiveness represent a node's available buffer storage for the new packets that are sent to it. In the above-mentioned formula, $B_c(x)$ is the node's buffer space capacity and $\sum_{i=1}^{N} m_{size}^{i}(x)$ is the summary of all message occupancies [68]. The trust value based on the probability of congestion can be calculated as follows:

$$T_{cong-free} = P_{cong-free} = \Big(\frac{m_{size}^{new}(x) - Ret(x)}{m_{size}^{new}(x)}\Big) \tag{5.12}$$

Congestion is estimated by measuring the free buffer space in the link layer. Therefore, only the nodes with sufficient buffer space forward the route request packet and participate in the routing process.

Malicious Packet Loss Detection

In the proposed malicious packet loss detection, as discussed earlier, the detection is based on cross-layer trust-based technique. The features are collected and extracted from physical, network and data link layer and they are used in the trust model generation to build a trust model based on legitimate causes of packet losses in the network. In order to detect the malicious packet losses, we first need to calculate the trust value that one node holds from all its neighbors based on the individual trust models.

In the first step, we try to derive a set of nodes that are suspicious to be malicious. However, in order to do that, we need to make sure that the node is trusted. Unlike many works that believe if the trust value falls below a predefined threshold, it indicates that the node is malicious, we claim that the if the trust value of one node from another node is less than threshold it only indicates that if the packet loss occurs, it is based on legitimate reasons and cannot determine existence of malicious node in the network. One main fact to support this idea is that, since the characteristics of malicious node is in most of the cases ideal for instance in terms of energy efficiency or transmission range or even it cannot be a victim of another malicious node causing disruption by collision, the node with low trust value is in contract with this notion due to the fact that trust model is based on residual energy, collision-free, congestion-free and link stability.

Therefore, if the trust value of one node from another node is less than a certain threshold, the node cannot be malicious. In fact, all other nodes with higher trust value than the threshold are suspicious to be the malicious node. This is the basis of creating a list of suspicious node that have the potential to be malicious in the network. In order to detect the malicious node

from the suspicious list, the number of received and forwarded packets are compared to determine whether a packet loss has occurred by a certain node or not. For cases where there is a difference between the forwarded packets and received packets and if the node which drops the packet is in the suspicious list, then it is certainly a malicious node that drops the packets deliberately. Algorithm 2 demonstrates the pseudo code for the proposed cross-layer trust-based IDS for malicious packet loss detection.

The detection of malicious node is important since they should be eliminated from the network and all the network-related operations. However, the node with low trust values should remain in the network. Although, the node with low trust value cannot participate in routing the transmitted packet to the destination, however, since the trust value of nodes get updated frequently, i.e. after every packet transmission, it may obtain a higher trust value.

5.3.3 Response Module

The list of suspicious nodes generated by the monitoring unit. This list is sent to all the neighboring nodes. This unit distributes the decision made by detection analysis engine with the list of intrusions to all nodes. This unit raises the flag to inform all the nodes in the network about the intrusive attack and the intruder node.

5.4 Proposed Secure Path Selection

Secure routes are the key to provide reliability and enhance the throughput of the network. After data collection about the network behavior from multiple layers, the data analysis module which represents the layer that we aim to detect an attack in, will process the data to provide

an informative decision about the normality of the network behavior. Based on proposed trust model, the trust value of every node is evaluated to help in the selection of the next hop for secure and reliable routing.

Using the provided trust values from the observer node to the source node and based on the secure trust-based routing model proposed, the source node calculates the secure selected path

Algorithm 2 Proposed Trust-based Cross-layer Malicious Packet Loss Detection Algorithm

Input: No. of forwarded packets and No. of received packets, N=No. of nodes in the network
Output: Malicious Node Detection
Step 1: ***Derive a suspicious list***
for $i = 1$ to N **do**
  for $j = 1$ to M *and* $j \neq i$ **do**
    Check neighbor table for node i
    Calculate Trust T_i^j for all neighbor nodes
    if $0 < T_i^j < Threshold$ **then return**
      Put node j in the not trusted list
    else if $T_i^j > Threshold$ **then return**
      Put node j in the suspicious list
    elsereturn
      Eliminate the node from neighbor nodes
    end if
  end for
end for
Step 1: ***Detect malicious node***
while $No.\ of\ recieved\ packets\ by\ j \neq No.\ of\ forwarded\ packets\ by\ i$ **do**
  if node j is in suspicious list **then return**
    node j is malicious
  elsereturn
    Eliminate trust value of node j
  end if
end while

as the most reliable and secure path for routing.

$$Path - Trust = \frac{\sum (TV.Min\ Value).\sqrt{No.\ of\ hops}}{No.\ of\ hops} \tag{5.13}$$

$$Selected\ Secure\ Path = max(path - Trust_i) \tag{5.14}$$

5.4.1 Observer Node Selection

In the proposed technique, the node with the highest absolute trust values will be selected as an observer node. The observer node plays an important role in the proposed routing technique. It is responsible for monitoring the nodes in the network, maintain and update the trust table of nodes and broadcast these trust values to the neighboring nodes. It also calculates the trust values of all possible routes to the destination. After every node in the network has the trust values of their neighboring nodes, the previous observer node assigns the node with the highest trust value as the next observer node. The responsibilities of the observer nodes are:

1.To monitor the network and calculate the trust values of the nodes

2.To update the table of the trust values of all the neighbor nodes

3.To broadcast the trust value of every node to source

4.To decrease the trust value of the nodes with malicious activity

5.5 Performance Evaluation

Network Simulator (NS2) is used for simulation which consists of the collection of network protocols to enable simulation of various topologies with different types protocols. The proposed routing technique is implemented in AODV through coding with C++.

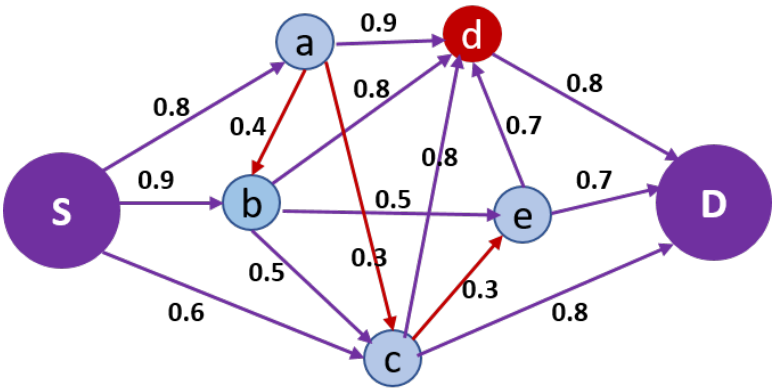

Figure 5.3: Secure Path Selection Based on Trust-based Cross-layer AODV

Table 5.3: Trust Table

	S	a	b	c	d	e	D
S		**0.8**	**0.9**	**0.6**			
a			**0.4**	**0.3**	**0.9**		
b				**0.5**	**0.8**	**0.5**	
c					**0.8**	**0.3**	**0.8**
d							**0.8**
e							**0.7**

5.5.1 Validation of Trust Value Evaluation

To better demonstrate the proposed trust-based routing path selection, an example of secure path selection from both malicious packet loss attack and faulty network that results in packet loss, is provided. As it can be observed from Fig. 3, there is a network consisting of a single transmitter and a receiver and there exist five nodes that relay the transmitted packet. Firstly, each node calculates the trust values of all its neighbor nodes as demonstrated in Fig. 5.3. In the second step, the compromised nodes that drops the packets are eliminated and all the links from other nodes to it would not be used for routing. Moreover, the links with overall trust values less than a threshold would not be used since these links are not reliable to relay the packets to the next hop. The route with the highest trust values will be used for routing in the network layer. At the start, the trust value of all nodes is equal to 1 and trust value changes due

to receiving or losing packets based on the trust value formula. After the transmission starts, the trust values will be updated in the network layer of AODV protocol. When the destination node receives RREQ, the updated trust value of each node will be sent to the source with RREP. Table 5.3 demonstrates the trust value between every two neighbor node. As it can be seen from both Figure 5.3 and table 5.3, based on the proposed algorithm, node d is detected to be a compromised node that drops the packets. Therefore, it is eliminated from the routing table(shown in red color). Moreover, since T_a^b and T_a^c and T_c^e have trust values less than the threshold (here $Thr = 0.5$), these links are also eliminated and not considered to be used in the routing path. Based on the remaining legitimate nodes and links with high trust values, there are three routing path available. In order to find the secure and reliable routing path, the path trust should be calculated for each path based on equation 5.13 and 5.14. Therefore we have:

$S \rightarrow b \rightarrow e \rightarrow D$ Path-Trust= 1.2

$S \rightarrow b \rightarrow c \rightarrow D$ Path-Trust= 1.27

$S \rightarrow c \rightarrow D$ Path-Trust= 0.3 based on the calculation of the path-trust values, the second routing path consisting of nodes S,b,c and D is the most secure and reliable path.

5.5.2 Performance Metrics

In our simulation, we have considered the packet start time, end time, number of generated, dropped and received packets as well as packet delivery ratio and false positive rate versus the percentage of malicious nodes to all nodes in the network. The main goal of this study is to improve the following metrics in intrusion detection systems:

1. Detection rate which is also known as true positive rate and represent the rate of correctly

identifies attacks. The performance of the proposed protocol is analyzed using a number of packet losses, packet delivery ratio and the number of received packets with a variation of the number of the nodes, speed.

2. False alarm rate or false positive rate that represents the rate of normal behavior which incorrectly identified as intrusion

It is notable that the Packet Delivery Ratio(PDR), routing overhead and average throughput has also been taken into considerations.

$$PDR = \frac{No.\ of\ recieved\ packets}{No.\ of\ transmitted\ packets} \tag{5.15}$$

5.6 Simulation Results

In this work, the proposed cross-layer design is compared to conventional single layer method and other cross-layer designs in terms of false positive rate.

5.6.1 Assumption

In this study, couple of assumptions are considered. First, it is assumed that malicious node never initiates route discovery since it does not have any data to transmit. In other words, an adversary node never forwards RREQ message. Moreover, it is assumed that the malicious node either drops the packets or forward it. The case where the malicious node forwards the modified and corrupted version of packets remains for future work. In this study, its is also assume that the number of malicious nodes is always less than legitimate nodes. In addition, we

Table 5.4: Network Configuration for Performance Analysis

Number of nodes	50
Simulation time	20 sec
Environment size	1000m x 1000m
Channel Type	Wireless Channel
Propagation Model	Two Ray Ground
MAC Type	IEEE 802.11
Transmission rage	180m
Routing Protocol	AODV
Antenna Type	Bidirectional
Packet Size	512 Byte
Mobility Speed	20 m/s

consider that all mobile nodes are deployed randomly and it is assumed that all the nodes have computing, storing and communication capability. A homogeneous system is considered where every node has the similar initial energy level. Furthermore, we assume that there are enough interactions between nodes that they only require direct trust and a third party to confirm the trustworthy of a node(indirect trust) is not necessary required to be obtained. We suppose that every node has a list of all its immediate neighbors and all immediate neighbors of a node have a trust value of 1 which indicates that a node fully trusts its neighboring nodes initially and after getting feedback from the behavior of the network, it will update the corresponding trust values. In addition, both the source and the destination node are not malicious. The attack model considered in this study does not consider the collaboration among attackers to cause a problem in the network.

5.6.2 MANET Routing Protocol

MANET routing protocols are classified as proactive and reactive protocols. Proactive protocols are not as productive as reactive protocols and that is the main reason why Ad hoc

On Demand Distance Vector(AODV) and Dynamic Source Routing(DSR) are mostly used in MANETs [44]. AODV is a reactive routing protocol which unlike active protocols that rely on routing table to establish a path to the destination, it finds a route to the destination whenever a communication is needed [38]. AODV belongs to distance vector class in which every node has knowledge about its neighbor nodes and the distance and cost to reach them. In AODV, every node sends the routing table to the neighbor nodes, to support the connectivity of the nodes in mobile ad-hoc networks, we used AODV(Ad-hoc On Demand Distance Vector) routing protocol is used in this study. In AODV, the relay nodes are responsible for finding a path to the destination and sending PREQ to the neighboring node just like the transmitter. AODV provide loop-free, self-starting and scalable network routing for ad hoc networks [76]. It is a very productive protocol due to having fewer overheads and are mostly used for MANETs [44]. This protocol builds routes based on the desire of the source node and the communication links depend on the sequence number of their corresponding nodes. Any changes in the network will change the sequence and help to better identification of the recent network activity. It is notable that the routes are maintained until they are no longer desired. In AODV four types of messages are used to communicate between nodes, namely: route request message(RREQ), route reply message(RREP), route error message(RERR) and route reply acknowledge message(RREP-Ack). Among which, the first two are used for route discovery while the rest are used for maintenance. Once the source wants to establish a connection to the destination, it starts route discovery phase. Finding a path to the destination is possible through broadcasting RREQ message to all the immediate neighbors by the source node. When a source node requests a route to a destination for which the route is unknown, it broadcasts PREQ to the network [4]. If

[4]It is notable that this will happen if the source node does not have any route in its route table.

a neighbor node finds a route through itself to the destination, it response with RREP packet. Otherwise, it forwards the received RREQ packet to the network in its transmission and allows other nodes to decide the path. Nodes which resides in the communication range of the source node will send RREP to the source node under two circumstances: First, if it is the destination, second if it has a route to the destination. Every node in the network keeps RREQ's source IP address and broadcast ID. Once the source receives the RREP it forwards packets to the node it received the RREP from. If any neighbor node could find a routing passing through itself, it replays back with RREP and if no neighbor node could find a path, then the source node will generate RREQ again. This process will continue until the path from source to the destination is established. During the data transmission, the source node updates its routing table based on multiple RREP that it receives [84]. Based on this study, when the source node receives an RREP message with secure selected path which differs from the previous path, it would revise the routing information for the same destination and start to forward the data packets with the new secure and reliable route. It is notable that AODV maintains the routing path information while there is periodic traffic between source and destination and would disregard the path as soon as it becomes inactive and the new route between source and destination become established [76]. If the link breaks, the source node will receive RRER to inform the source node about the broken link. The source node will do the process of route discovery after it receives RERR for the active route. In AODV, every node has its own routing table, all the nodes in the network, the distance to the next hop. Every node sends to the whole routing table to the neighbor nodes [38]. If the source node later receives an RREP message which has fewer hop counts or has better performance for routing it revises its routing table and forwards packets using the better route. As long as the route is active, there is periodic data traffic between the source and

destination, it maintains routes in the routing table otherwise it becomes inactive and deletes the route. Each node contains a list of neighboring nodes to which it can communicate directly through a "HELLO" message and it uses link layer to update the list and keep track of neighboring nodes to avoid resources consumption such as energy and bandwidth [82]. In AODV, shortness of the path and freshness of the path which is readable through RREP message are the important metrics to find the best route to the destination [83].

5.6.3 Simulation Environment

In this work, we have used the malicious nodes that drop the packets in simulated MANET environment. All the simulations are conducted using Network Simulator(NS2) that have a collection of all network protocols. It is an object oriented, event driven network simulator written in C++ and OTcl. The simulation configuration as can be seen from table IIIconsist of 15, 20, 40 and 50 nodes distributed over a 1000m x 1000m area. The simulation was carried for 50 seconds and packets are transmitted with constant bit rate(CBR) of 5 packets per second. In order to determine the different behavior of the network, we have carried out the simulation with different parameters. The hardware configuration used for the simulation are shown in table 5.4.

Detection Performance

To evaluate the detection accuracy of the proposed trust-based cross-layer IDS, the performance of 5 packet loss detection including proposed technique in terms of false positive rate with respect to the increase in the percentage of malicious node to the total number of nodes in the network is studied in figure 5.4. It is clear that with the increase in the number of malicious

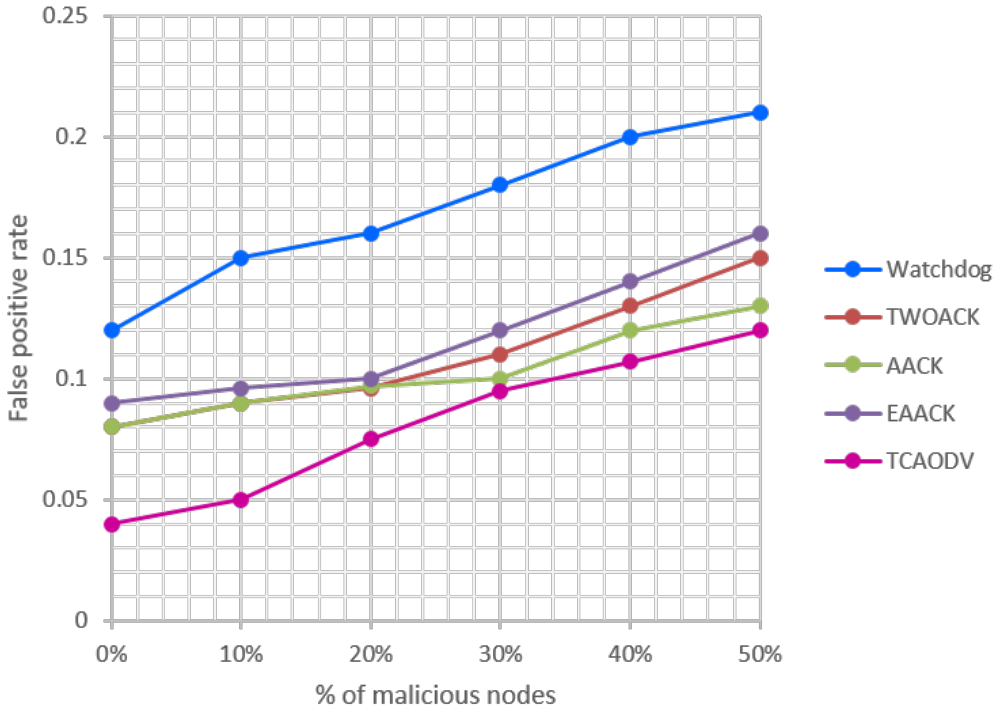

Figure 5.4: Performance of Proposed Trust-based Cross-layer IDS in terms of False Positive Rate

nodes, the accuracy will decrease and therefore, the detection system will have more false positive rate. From Fig 5.4, it can be observed that the proposed packet loss detection technique has significantly less positive rate than conventional watchdog mechanism. Moreover, comparing to other techniques which are based on ACK packet, namely: TWOACK,AACK,EAACK, trust-based cross-layer IDS outperforms other packet loss detection techniques. It is notable that the significance of TCAODV is not limited to more accurate decision. The proposed technique is more desirable since it can distinguish malicious packet loss from faulty network packet losses which is extremely important.

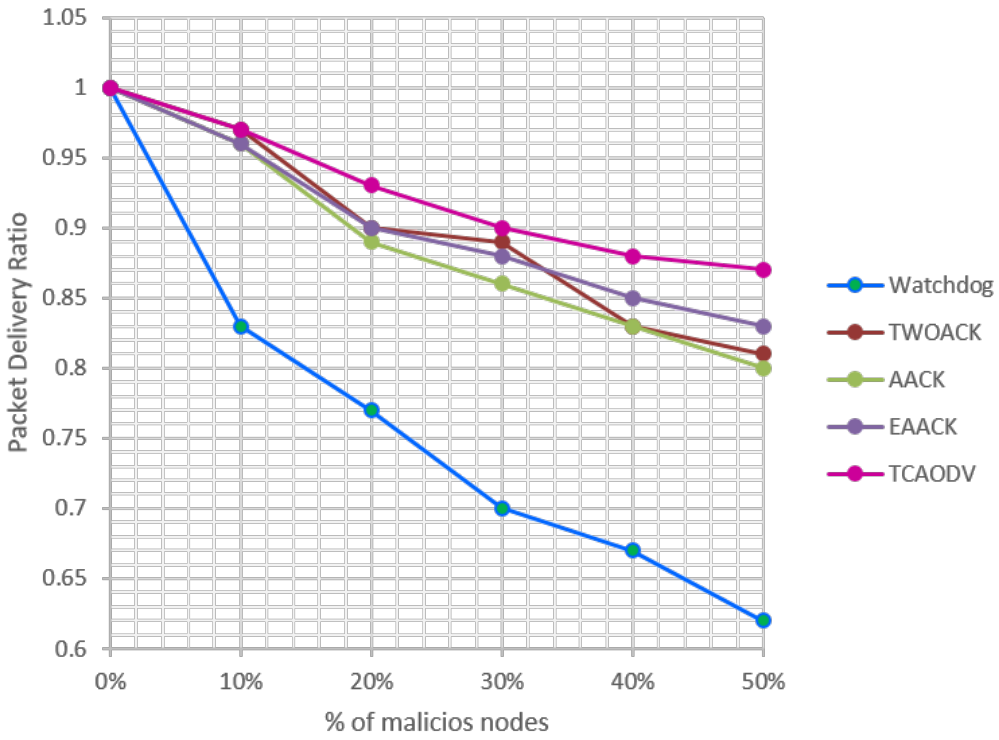

Figure 5.5: Comparison of Packet Delivery Ratio of Proposed Trust-based Cross-layer IDS with Other Packet Loss Detection Technique

Secure Path Performance

Since detection of an adversary node (here, the node that maliciously drops the packets)is not enough to secure the network from attacks that threaten the network, the network should be smart enough to not use such nodes as relay on the path from source to destination. Therefore, it is important to evaluate the performance of the detection mechanism in providing more secure network and study the secure packet delivery ratio as well. In this regard, the performance of the proposed malicious packet loss detection mechanism is studied in terms of packet delivery ration with respect to percentage of the number of malicious nodes to the total number of nodes in the network. Moreover, the packet delivery ration of proposed technique is compared to related works in the literature. As Fig 5.5 demonstrates, fewer number of packets will

be delivered as the number of malicious node grows because the accuracy of any detection technique decreases with more number of malicious nodes and therefore, there might be some malicious nodes that are used as relay as hence, the network experience low packet delivery ratio. Moreover, Fig. 5.5 demonstrate that the proposed technique outperforms other packet loss detection mechanism in terms of packet delivery ratio.

5.7 Summary

Detection-based approaches have attracted lots of attentions in the last few years since prevention techniques for security vulnerability such as authentication and key-management schemes are not capable of identifying and defending the insider attacks effectively. Therefore, the need for misbehavior detection plays a vital role in wireless networks. Among all the mitigation techniques in the literature, intrusion detections have gained lots of attention among researchers due to their significance ability and performance. On the other hand, not all IDS can provide accurate malicious behavior detection. Single layer IDS has significantly lower detection rate and accuracy comparing to cross-layer IDSs since they use attributes and features from various layers of the protocol to make a decision about the performance of a node in the network. In this study, we are interested in identifying the packet loss attack in the network. We considered different causes of packet loss to reduce false positive rate in identifying malicious packet loss in the network better. Since packet loss attack happens in the network and can severely affect the performance of routing in the net- work, the well-known approach to secure network layer to provide more reliable and secure routing path is used in this study. Effective packet loss detection and providing secure and reliable routing path in the network

is proposed in this study using cross-layer trust-based detection system which is based on the AODV protocol. This work investigated harmful Denial of Services attack that is represented in different forms across the different layer of the protocol stack. The main scope of this study is to demonstrate the strength of the cross-layer approach to single layer intrusion detection design. We have considered non-legitimate reasons for possible packet loss in the network to determine malicious and non-malicious packet losses. Moreover, different contributing factors to packet loss have been considered and used in this study. We showed how cross-layer design can secure the communication between nodes and provide a secure traffic route between transmitter and receiver. The simulation conducted in NS2 illustrate that proposed scheme achieves high performance in terms of false positive rate. The only drawback of the proposed scheme is that it can be easily tricked by an on-off attacker in which the malicious node would act normal in cases its trust value is significantly low to gain the trust from other neighboring nodes and once its trust value obtains a certain level, it would act as a malicious node again. The study for this attack remains for future works. The major drawback of this approach is that based on its trust evaluation and detection mechanism, it is incapable of detection of the modified packet transmission. Moreover, even though this approach is very effective in terms of the performance of the network however with the cost of more overhead.

Chapter 6

Conclusion and Future Works

6.1 Conclusion

In the first chapter, motivation behind this study is presented. Security mechanism is one of the fundamental parts of any wireless networks since many, if not all, of the application rely on receiving a secure message. Therefore, the need for security of data transmission is undeniable.In this study, we studied different aspects of detection techniques and focused on the intrusion detection techniques specifically. IDS is a system that monitors and analyses the network traffic behavior to identify any abnormality behavior in the network. There are many challenges in design and implementation of such security systems that each chapter of work focused on proposing an efficient solution to. Chapter II presented a comprehensive literature review of existing studies in the area of security mechanisms for wireless networks. A vast number of techniques are studied and their advantages and drawbacks are reviewed in this chapter.

In the third chapter; however, a feature selection technique to derive the informative at-

tributes about the network behavior is proposed. One of the major issue with current intrusion detection technique is the huge number of data traffic instances with many attributes which all are not necessary required for analyzing the network traffic behavior for security purposes. Therefore, the raw data should be processed and only a set of informative network traffic features to the detection engine should be derive. In chapter III a correlation-based feature selection technique for intrusion detection systems is proposed. We showed that in order to determining good features, along with considering their high correlation to class labels, and least inter correlations, the features should fit well in the selected subset and considering the inter correlation among features after the selection of the candidate subset is necessary and corresponding selected features are the best features contributing to discrimination of network behavior. Correlation-based feature selection(CFS) and symmetrical uncertainty(SU) are the two dependency metrics used in this paper. The proposed feature selection technique is compared with other well-known feature selection algorithms namely: CFS, IG, GR and chi-squared on NSL-KDD dataset. The results indicate that the proposed technique has considerably less training time while maintaining accuracy and precision. In addition, different feature selection techniques are tested with different classifiers in terms of detection rate and FAR. Regardless of the classification algorithm, the results indicate that the proposed scheme out performs other techniques. Another observation from comparison results between the proposed technique and using the full dataset is that J48 classification algorithm performs better with proposed feature selection algorithm than other classifiers.

In the fourth chapter, two well-known classification algorithms are presented to detect misbehavior in wireless networks. The performance of support vector machine(SVM) and relevance vector machine(RVM) are illustrated and compared using feature selection techniques

and without considering attribute selection. Moreover, the affect of different kernel functions, that maps the data into the another dimension for better classification, is studied. Since RVM relies on the less number of vectors to classify the instances into normal or anomaly categories and hence requires less training time. On the other hand, despite the fact that SVM needs more number of support vectors for classification, it demonstrate to have better performance.

On the other hand, not all IDS can provide accurate malicious behavior detection. Single layer IDS has significantly lower detection rate and accuracy comparing to cross-layer IDSs since they use attributes and features from various layers of the protocol to make a decision about the performance of a node in the network. Chapter V studied identifying the packet loss attack in the network using multi-layer features based on trust model. We have considered non-legitimate reasons for possible packet loss in the network to determine malicious and non-malicious packet losses to reduce false positive rate. Moreover, different contributing factors to packet loss have been considered and used in this study. Since packet loss attack happens in the network and can severely affect the performance of routing in the network, the well-known approach to secure network layer to provide more reliable and secure routing path is used in this study. Effective packet loss detection and providing secure and reliable routing path in the network is proposed in this study using cross-layer trust-based detection system which is based on the AODV protocol. Moreover, we showed how cross-layer design can secure the communication between nodes and provide a secure traffic route between transmitter and receiver.

6.2 Future Works

IDS plays a vital role in securing the network from attacks due to its ability in detecting and identifying attacks accurately and quickly which makes it a very important area of research. This dissertation focused on efficiency enhancement of IDS and its applicability on real wireless networks, here MANET. However, the study in IDS design will require further investigation in future.

One possible direction for future studies is that since there is always a trade-off between the performance accuracy and efficiency of intrusion detection systems, and also because difference wireless network systems with different application require security systems with different security level, building a statistical model of intrusion detection with feature selection and detection strategy to meet the required performance.

Moreover, it is important to consider more features from network protocol and engage upper layers, e.g. application layer to detect malicious packet loss attack in the network. Considering more layers and associated features will increase the accuracy of detection intrusive nodes in the network. Although detection of the malicious packet loss is extremely important to secure the routing path from source to destination, the importance of other malicious activities and other attacks in the network should not be neglected. For example, one possible direction for future works can be investigation on design of an accurate IDS to detect the modified packet transmission since this study only covered the identification of the cases where the data packets are lost but not modified. Moreover, the information that can be obtained from other layers of the protocol stack, e.g. transport layer or application layer can be studied and use their features in better detection of attackers and malicious nodes and provide more secure networks.

In addition to the above-mentioned area of research for future work, one can consider the existence of intrusion detection system design in every node of a network and use the collaboration among nodes to make decision, first, about the existence of an intruder and second to identify the intrusive node. Game theory as one of the very powerful techniques in decision collaborative environments can be adopted for the cooperation of IDS from all or selected nodes in the network for making decision about the normality or abnormality behavior of the network.

Bibliography

[1] NSL-KDD data set for network-based intrusion setection systems.

[2] Waikato environment for knowledge analysis (WEKA) version 3.7.13.

[3] Carrads: Cross layer based adaptive real-time routing attack detection system for {MANETS}. *Computer Networks*, 54(7):1126 – 1141, 2010.

[4] A. K. Abdelaziz, M. Nafaa, and G. Salim. Survey of routing attacks and countermeasures in mobile ad hoc networks. In *2013 UKSim 15th International Conference on Computer Modelling and Simulation*, pages 693–698, April 2013.

[5] A. Abduvaliyev, A. S. K. Pathan, J. Zhou, R. Roman, and W. C. Wong. On the vital areas of intrusion detection systems in wireless sensor networks. *IEEE Communications Surveys Tutorials*, 15(3):1223–1237, Third 2013.

[6] A. A. Aburomman and M. Bin Ibne Reaz. Survey of learning methods in intrusion detection systems. In *2016 International Conference on Advances in Electrical, Electronic and Systems Engineering (ICAEES)*, pages 362–365, Nov 2016.

[7] A. Al-Roubaiey, T. Sheltami, A. Mahmoud, E. Shakshuki, and H. Mouftah. Aack: Adaptive acknowledgment intrusion detection for manet with node detection enhancement. In *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, pages 634–640, April 2010.

[8] M. M. Alani. Manet security: A survey. In *2014 IEEE International Conference on Control System, Computing and Engineering (ICCSCE 2014)*, pages 559–564, Nov 2014.

[9] Mohammed A. Ambusaidi, Xiangjian He, Priyadarsi Nanda, and Zhiyuan Tan. Building an intrusion detection system using a filter-based feature selection algorithm. PP(99):1–13, 2016.

[10] Fatemeh Amiri, MohammadMahdi Rezaei Yousefi, Caro Lucas, Azadeh Shakery, and Nasser Yazdani. Mutual information-based feature selection for intrusion detection systems. *Journal of Network and Computer Applications*, 34(4):1184 – 1199, 2011.

[11] James P Anderson et al. Computer security threat monitoring and surveillance. Technical report, Technical report, James P. Anderson Company, Fort Washington, Pennsylvania, 1980.

[12] F. J. Aparicio-Navarro, K. G. Kyriakopoulos, and D. J. Parish. An on-line wireless attack detection system using multi-layer data fusion. In *2011 IEEE International Workshop on Measurements and Networking Proceedings (M N)*, pages 1–5, Oct 2011.

[13] R. Baiad, H. Otrok, S. Muhaidat, and J. Bentahar. Cooperative cross layer detection for blackhole attack in vanet-olsr. In *2014 International Wireless Communications and Mobile Computing Conference (IWCMC)*, pages 863–868, Aug 2014.

[14] K. Balakrishnan, Jing Deng, and V. K. Varshney. Twoack: preventing selfishness in mobile ad hoc networks. In *IEEE Wireless Communications and Networking Conference, 2005*, volume 4, pages 2137–2142 Vol. 4, March 2005.

[15] D. Bansal and S. Sofat. Use of cross layer interactions for detecting denial of service attacks in wmn. In *2010 14th International Telecommunications Network Strategy and Planning Symposium (NETWORKS)*, pages 1–6, Sept 2010.

[16] D. Bansal, S. Sofat, and P. Kumar. Distributed cross layer approach for detecting multilayer attacks in wireless multi-hop networks. In *2011 IEEE Symposium on Computers Informatics*, pages 692–698, March 2011.

[17] R. Bansal, N. Gaur, and S. N. Singh. Outlier detection: Applications and techniques in data mining. In *2016 6th International Conference - Cloud System and Big Data Engineering (Confluence)*, pages 373–377, Jan 2016.

[18] B. I. A. Barry and H. A. Chan. Towards intelligent cross protocol intrusion detection in the next generation networks based on protocol anomaly detection. In *The 9th International Conference on Advanced Communication Technology*, volume 3, pages 1505–1510, Feb 2007.

[19] Khalid Benabdeslem and Mohammed Hindawi. Efficient semi-supervised feature selection: Constraint, relevance, and redundancy. 26(5):1131–1143, May 2014.

[20] PK Bhakthavatsalam and B Malarkodi. Securing provenance by avoiding packet drop attack due to congestion in wireless sensor network. In *Inventive Computation Technologies (ICICT), International Conference on*, volume 2, pages 1–4. IEEE, 2016.

[21] S. Bose and A. Kannan. Detecting denial of service attacks using cross layer based intrusion detection system in wireless ad hoc networks. In *2008 International Conference on Signal Processing, Communications and Networking*, pages 182–188, Jan 2008.

[22] A. L. Buczak and E. Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. *IEEE Communications Surveys Tutorials*, 18(2):1153–1176, Secondquarter 2016.

[23] Anna L. Buczak and Erhan Guven. A survey of data mining and machine learning methods for cyber security intrusion detection. 18(2):1153–1176, 2016.

[24] O. Can and O. K. Sahingoz. A survey of intrusion detection systems in wireless sensor networks. In *2015 6th International Conference on Modeling, Simulation, and Applied Optimization (ICMSAO)*, pages 1–6, May 2015.

[25] S. S. Chakravarthi and S. Veluru. A review on intrusion detection techniques and intrusion detection systems in manets. In *2014 International Conference on Computational Intelligence and Communication Networks*, pages 730–737, Nov 2014.

[26] N. Chand, P. Mishra, C. R. Krishna, E. S. Pilli, and M. C. Govil. A comparative analysis of svm and its stacking with other classification algorithm for intrusion detection. In *2016 International Conference on Advances in Computing, Communication, Automation (ICACCA) (Spring)*, pages 1–6, April 2016.

[27] Girish Chandrashekar and Ferat Sahin. A survey on feature selection methods. *Computers and Electrical Engineering*, 40(1):16 – 28, 2014.

[28] R. Chitrakar and H. Chuanhe. Anomaly detection using support vector machine classification with k-medoids clustering. In *2012 Third Asian Himalayas International Conference on Internet*, pages 1–5, Nov 2012.

[29] Te-Shun Chou, Kang K. Yen, Jun Luo, Niki Pissinou, and Kia Makki. Correlation-based feature selection for intrusion detection design. In *Military Communications Conference (MILCOM)*, pages 1–7, October 2007.

[30] R. Chourasia and R. K. Boghey. Novel ids security against attacker routing misbehavior of packet dropping in manet. In *2017 7th International Conference on Cloud Computing, Data Science Engineering - Confluence*, pages 456–460, Jan 2017.

[31] A. C. Enache and V. Sgârciu. Anomaly intrusions detection based on support vector machines with bat algorithm. In *2014 18th International Conference on System Theory, Control and Computing (ICSTCC)*, pages 856–861, Oct 2014.

[32] Pablo A. Estévez, Michel Tesmer, Claudio A. Perez, and Jacek M. Zurada. Normalized mutual information feature selection. 20(2):189–201, 2009.

[33] L. Gandhimathi and G. Murugaboopathi. Cross layer intrusion detection and prevention of multiple attacks in wireless sensor network using mobile agent. In *2016 International Conference on Information Communication and Embedded Systems (ICICES)*, pages 1–5, Feb 2016.

[34] Tanya Garg and Ya Kumar. Combinational feature selection approach for network intrusion detection system. In *Int. Conf. on Parallel, Distributed and Gride Computing (PDGC)*, pages 82–87, December 2014.

[35] Yi Gong, Yong Fang, Liang Liu, and Juan Li. Multi-agent intrusion detection system using feature selection approach. In *10th Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing (IIH-MSP)*, pages 528–531, August 2014.

[36] D. Grochocki, J. H. Huh, R. Berthier, R. Bobba, W. H. Sanders, A. A. Cárdenas, and J. G. Jetcheva. Ami threats, intrusion detection requirements and deployment recommendations. In *2012 IEEE Third International Conference on Smart Grid Communications (SmartGridComm)*, pages 395–400, Nov 2012.

[37] Y. Guang and N. Min. Anomaly intrusion detection based on wavelet kernel ls-svm. In *Proceedings of 2013 3rd International Conference on Computer Science and Network Technology*, pages 434–437, Oct 2013.

[38] N. K. Gupta and K. Pandey. Trust based ad-hoc on demand routing protocol for manet. In *2013 Sixth International Conference on Contemporary Computing (IC3)*, pages 225–231, Aug 2013.

[39] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann, and I. H. Witten. The weka data mining software: an update. *ACM SIGKDD Explorations Newsletter*, 11(1):10–18, 2009.

[40] Mark A. Hall. *Correlation-based Feature Selection for Machine Learning*. PhD thesis, Univeristy of Waikato, Hamilton, New Zealand, 1999.

[41] Mark A. Hall and Geoffrey Holmes. Benchmarking attribute selection techniques for discrete class data mining. 15(6):1437 – 1447, 2003.

[42] T. Hayajneh, P. Krishnamurthy, D. Tipper, and T. Kim. Detecting malicious packet dropping in the presence of collisions and channel errors in wireless ad hoc networks. In *2009 IEEE International Conference on Communications*, pages 1–6, June 2009.

[43] S Hemalatha, PC Senthil Mahesh, Paul Rodrigues, and M Raveendiran. Analysing cross layer performance based on sinking and collision behaviour attack in manet. In *Radar, Communication and Computing (ICRCC), 2012 International Conference on*, pages 77–82, March 2012.

[44] K.Palanivel I.Meenatchi. Cross layer intrusion detection in manet's: A survey. *International Journal of Advanced Research in Computer Engineering & Technology (IJARCET)*, 4(3):622–626, 2015.

[45] C. Ioannou and V. Vassiliou. The impact of network layer attacks in wireless sensor networks. In *2016 International Workshop on Secure Internet of Things (SIoT)*, pages 20–28, Sept 2016.

[46] J. Jiang, G. Han, F. Wang, L. Shu, and M. Guizani. An efficient distributed trust model for wireless sensor networks. *IEEE Transactions on Parallel and Distributed Systems*, 26(5):1228–1237, May 2015.

[47] J. Jiang, X. Jing, B. Lv, and M. Li. A novel multi-classification intrusion detection model based on relevance vector machine. In *2015 11th International Conference on Computational Intelligence and Security (CIS)*, pages 303–307, Dec 2015.

[48] J. Jiang, X. Jing, B. Lv, and M. Li. A novel multi-classification intrusion detection model based on relevance vector machine. In *2015 11th International Conference on Computational Intelligence and Security (CIS)*, pages 303–307, Dec 2015.

[49] J. F. C. Joseph, A. Das, B. C. Seet, and B. S. Lee. Cross layer versus single layer approaches for intrusion detection in manets. In *2007 15th IEEE International Conference on Networks*, pages 194–199, Nov 2007.

[50] P. Joshi, P. Nande, A. Pawar, P. Shinde, and R. Umbare. Eaack - a secure intrusion detection and prevention system for manets. In *2015 International Conference on Pervasive Computing (ICPC)*, pages 1–6, Jan 2015.

[51] K. Jothilakshmi, G. Usha, and S. Bose. A framework of cross layer based anomaly intrusion detection and self healing model for manet. In *2013 International Conference on Recent Trends in Information Technology (ICRTIT)*, pages 429–433, July 2013.

[52] M. Jouad, S. Diouani, H. Houmani, and A. Zaki. Security challenges in intrusion detection. In *2015 International Conference on Cloud Technologies and Applications (CloudTech)*, pages 1–11, June 2015.

[53] M. E. Kabir and J. Hu. A statistical framework for intrusion detection system. In *2014 11th International Conference on Fuzzy Systems and Knowledge Discovery (FSKD)*, pages 941–946, Aug 2014.

[54] Rajveer Kaur, Gulshan Kumar, and Krishan Kumar. A comparative study of feature selection techniques for intrusion detection. In *Computing for Sustainable Global Development (INDIACom)*, pages 2120–2124, March 2015.

[55] Rajveer Kaur, Gulshan Kumar, and Krishan Kumar. A comparative study of feature selection techniques for intrusion detection. In *Computing for Sustainable Global Development (INDIACom),*, pages 2120 – 2124, November 2015.

[56] M. V. Kotpalliwar and R. Wajgi. Classification of attacks using support vector machine (svm) on kddcup'99 ids database. In *2015 Fifth International Conference on Communication Systems and Network Technologies*, pages 987–990, April 2015.

[57] L. Li, S. Ma, and Y. Zhang. Optimization algorithm based on genetic support vector machine model. In *2014 Seventh International Symposium on Computational Intelligence and Design*, volume 1, pages 307–310, Dec 2014.

[58] R. Li, X. Wang, L. Lei, and Z. Zhao. Probabilities modeling of multi-class based on relevance vector machine. In *2016 12th World Congress on Intelligent Control and Automation (WCICA)*, pages 2755–2759, June 2016.

[59] M. A. Manzoor and Y. Morgan. Real-time support vector machine based network intrusion detection system using apache storm. In *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pages 1–5, Oct 2016.

[60] T. K. Marve and N. U. Sambhe. A review on cross layer intrusion detection system in wireless ad hoc network. In *2015 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, pages 1–4, March 2015.

[61] S. Mukherjee, M. Chattopadhyay, and S. Chattopadhyay. A novel encounter based trust evaluation for aodv routing in manet. In *2015 Applications and Innovations in Mobile Computing (AIMoC)*, pages 141–145, Feb 2015.

[62] A. Murali and M. Rao. A survey on intrusion detection approaches. In *2005 International Conference on Information and Communication Technologies*, pages 233–240, Aug 2005.

[63] M. Odstrcil, A. Murari, and J. Mlynar. Comparison of advanced machine learning tools for disruption prediction and disruption studies. *IEEE Transactions on Plasma Science*, 41(7):1751–1759, July 2013.

[64] Stefano Paris, Cristina Nita-Rotaru, Fabio Martignon, and Antonio Capone. Cross-layer metrics for reliable routing in wireless mesh networks. *IEEE/ACM Transactions on Networking (TON)*, 21(3):1003–1016, 2013.

[65] Jim Parker, Anand Patwardhan, and Anupam Joshi. Cross-layer analysis for detecting wireless misbehavior. In *Proceedings of the IEEE Consumer Communications and Networking Conference (CCNC 2006)*, volume 1, pages 6–9, 2006.

[66] B. D. Patel and A. D. Patel. A trust based solution for detection of network layer attacks in sensor networks. In *2016 International Conference on Micro-Electronics and Telecommunication Engineering (ICMETE)*, pages 121–126, Sept 2016.

[67] Hanchuan Peng, Fuhui Long, and Chris Ding. Feature selection based on mutual information criteria of max-dependency, max-relevance, and min-redundancy. 27(8):1226–1238, August 2005.

[68] Milena Radenkovic and Andrew Grundy. Efficient and adaptive congestion control for heterogeneous delay-tolerant networks. *Ad Hoc Networks*, 10(7):1322–1345, 2012.

[69] Muhammad Rafi and Mohammad Shahid Shaikh. A comparison of svm and rvm for document classification. *arXiv preprint arXiv:1301.2785*, 2013.

[70] R. R. Reddy, Y. Ramadevi, and K. V. N. Sunitha. Effective discriminant function for intrusion detection using svm. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1148–1153, Sept 2016.

[71] F. Sabahi and A. Movaghar. Intrusion detection: A survey. In *2008 Third International Conference on Systems and Networks Communications*, pages 23–26, Oct 2008.

[72] R. SakilaAnnarasi and S. Sivanesh. A secure intrusion detection system for manets. In *2014 IEEE International Conference on Advanced Communications, Control and Computing Technologies*, pages 1174–1178, May 2014.

[73] P. Satam. Cross layer anomaly based intrusion detection system. In *2015 IEEE International Conference on Self-Adaptive and Self-Organizing Systems Workshops*, pages 157–161, Sept 2015.

[74] M. B. Shahbaz, Xianbin Wang, A. Behnad, and J. Samarabandu. On efficiency enhancement of the correlation-based feature selection for intrusion detection systems. In *2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, pages 1–7, Oct 2016.

[75] S. Shanthi and E. G. Rajan. Comprehensive analysis of security attacks and intrusion detection system in wireless sensor networks. In *2016 2nd International Conference on Next Generation Computing Technologies (NGCT)*, pages 426–431, Oct 2016.

[76] Sandeep Sharma and Rajesh Mishra. A cross layer approach for intrusion detection in manets. *International Journal of Computer APPLICATIONS*, 93(9):34–41, 2014.

[77] Y. Shi, F. Xiong, R. Xiu, and Y. Liu. A comparative study of relevant vector machine and support vector machine in uncertainty analysis. In *2013 International Conference on Quality, Reliability, Risk, Maintenance, and Safety Engineering (QR2MSE)*, pages 469–472, July 2013.

[78] Ali Shiravi, Hadi Shiravi, Mahbod Tavallaee, and Ali A. Ghorbani. Toward developing a systematic approach to generate benchmark datasets for intrusion detection. *Computers & Security*, 31(3):357 – 374, 2012.

[79] Mehrnoush Barani Shirzad and Mohammad Reza Keyvanpour. A feature selection method based on minimum redundancy maximum relevance for learning to rank. In *AI and Robotics (IRANOPEN)*, pages 1–5, April 2015.

[80] R. Shrestha, K. H. Han, D. Y. Choi, and S. J. Han. A novel cross layer intrusion detection system in manet. In *2010 24th IEEE International Conference on Advanced Information Networking and Applications*, pages 647–654, April 2010.

[81] Raman Singh, Harish Kumar, and R. K. Singla. Analysis of feature selection techniques for network traffic dataset. In *Int. Conf. on Machine Intelligence and Research Advancement (ICMIRA)*, pages 42–46, December 2013.

[82] L. S'nchez-Casado, G. Maci'-Fern'ndez, and P. García-Teodoro. An efficient cross-layer approach for malicious packet dropping detection in manets. In *2012 IEEE 11th International Conference on Trust, Security and Privacy in Computing and Communications*, pages 231–238, June 2012.

[83] M. T. Soleimani and M. Kahvand. Defending packet dropping attacks based on dynamic trust model in wireless ad hoc networks. In *MELECON 2014 - 2014 17th IEEE Mediterranean Electrotechnical Conference*, pages 362–366, 2014.

[84] N. Soliyal and H. S. Bhadauria. Preventing packet dropping attack on aodv based routing in mobile ad-hoc manet. In *2016 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, pages 1371–1375, Sept 2016.

[85] Guanghui Song, Jiankang Guo, and Yan Nie. An intrusion detection method based on multiple kernel support vector machine. In *Network Computing and Information Security (NCIS), 2011 International Conference on*, volume 2, pages 119–123. IEEE, 2011.

[86] Qinbao Song, Jingjie Ni, and Guangtao Wan. A fast clustering-based feature subset selection algorithm for high-dimensional data. 25(1):1–14, 2013.

[87] S. S. Soniya and S. M. C. Vigila. Intrusion detection system: Classification and techniques. In *2016 International Conference on Circuit, Power and Computing Technologies (ICCPCT)*, pages 1–7, March 2016.

[88] Pakkurthi Srinivasu, P. S. Avadhani, Suresh Chandra Satapathy, and Tummala Pradeep. A modified kolmogorov-smirnov correlation based filter algorithm for feature selection. In *Proc. of the Int. Conf. on Information Systems Design and Intelligent Applications*, pages 819–826, 2012.

[89] Mahbod Tavallaee, Ebrahim Bagheri, Wei Lu, and Ali A. Ghorbani. A detailed analysis of the kdd cup 99 data set. In *IEEE Symposium on Computational Intelligence for Security and Defense Applications*, pages 1–6, July 2009.

[90] G. Thamilarasu, A. Balasubramanian, S. Mishra, and R. Sridhar. A cross-layer based intrusion detection approach for wireless ad hoc networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005.*, pages 7 pp.–861, Nov 2005.

[91] G. Thamilarasu, A. Balasubramanian, S. Mishra, and R. Sridhar. A cross-layer based intrusion detection approach for wireless ad hoc networks. In *IEEE International Conference on Mobile Adhoc and Sensor Systems Conference, 2005.*, pages 7 pp.–861, Nov 2005.

[92] G. Thamilarasu and R. Sridhar. Exploring cross-layer techniques for security: Challenges and opportunities in wireless networks. In *MILCOM 2007 - IEEE Military Communications Conference*, pages 1–6, Oct 2007.

[93] Michael E Tipping. Sparse bayesian learning and the relevance vector machine. *Journal of machine learning research*, 1(Jun):211–244, 2001.

[94] W. Tong, L. Lu, Z. Li, J. Lin, and X. Jin. A survey on intrusion detection system for advanced metering infrastructure. In *2016 Sixth International Conference on Instrumentation Measurement, Computer, Communication and Control (IMCCC)*, pages 33–37, July 2016.

[95] Dimitris G Tzikas, Liyang Wei, Aristidis Likas, Yongyi Yang, and Nikolas P Galatsanos. A tutorial on relevance vector machines for regression and classification with applications. *EURASIP News Letter*, 17(2):4, 2006.

[96] M. Vidhya. Efficient classification of portscan attacks using support vector machine. In *2013 International Conference on Green High Performance Computing (ICGHPC)*, pages 1–5, March 2013.

[97] La The Vinh, Nguyen Duc Thang, and Young-Koo Lee. An improved maximum relevance and minimum redundancy selection algorithm based on normalized mutual information. In *Int. Symp. on Applications and the Internet (SAINT)*, pages 395–398, July 2010.

[98] X. Wang, J. S. Wong, F. Stanley, and S. Basu. Cross-layer based anomaly detection in wireless mesh networks. In *2009 Ninth Annual International Symposium on Applications and the Internet*, pages 9–15, July 2009.

[99] Z. Wei, H. Tang, F. R. Yu, M. Wang, and P. Mason. Trust establishment with data fusion for secure routing in manets. In *2014 IEEE International Conference on Communications (ICC)*, pages 671–676, June 2014.

[100] André Weimerskirch and Gilles Thonet. A distributed light-weight authentication model for ad-hoc networks. In *International Conference on Information Security and Cryptology*, pages 341–354. Springer, 2001.

[101] Xu Xiang-min, Mao Yun-feng, Xiong Jia-ni, and Zhou Feng-le. Classification performance comparison between rvm and svm. In *Anti-counterfeiting, Security, Identification, 2007 IEEE International Workshop on*, pages 208–211. IEEE, 2007.

[102] Q. Yang, H. Fu, and T. Zhu. An optimization method for parameters of svm in network intrusion detection system. In *2016 International Conference on Distributed Computing in Sensor Systems (DCOSS)*, pages 136–142, May 2016.

[103] Yang Yang and Wang Yu. Rvm assessment model for computer network attack. In *Computer Science and Information Technology (ICCSIT), 2010 3rd IEEE International Conference on*, volume 3, pages 104–107. IEEE, 2010.

[104] LI Yin, Ma Xingfei, Yang Mengxi, Zhao Wei, and Gu Wenqiang. Improved feature selection based on normalized mutual information. In *International Symposium on Distributed Computing and Applications for Business Engineering and Science (DCABES)*, pages 518–522, 2015.

[105] Lei Yu and Huan Liu. Feature selection for high-dimensional data: A fast correlation-based filter solution. In *Int'l Conf. Machine Leaning*, pages 856–863, 2003.

[106] Wei Miao Yu, Tiehua Du, and Kah Bin Lim. Comparison of the support vector machine and relevant vector machine in regression and classification problems. In *Control, Automation, Robotics and Vision Conference, 2004. ICARCV 2004 8th*, volume 2, pages 1309–1314. IEEE, 2004.

[107] M. Zhang, B. Xu, and J. Gong. An anomaly detection model based on one-class svm to detect network intrusions. In *2015 11th International Conference on Mobile Ad-hoc and Sensor Networks (MSN)*, pages 102–107, Dec 2015.

Appendices

Appendix A - NSL-KDD Dataset

Feature Description in NSL-KDD Dataset

Index	Feature Name	Description	Type
1	duration	length of the connection	continuous
2	protocol-type	type of the protocol, e.g. tcp, udp, etc	symbolic
3	service	network service on the destination, e.g., http, telnet, etc	symbolic
4	flag	normal or error status of the connection	symbolic
5	src-bytes	number of data bytes from source to destination	continuous
6	dst-bytes	number of data bytes from destination to source	continuous
7	land	1 if connection is from/to the same host/port; 0 otherwise	symbolic
8	wrong-fragment	number of “wrong” fragments	continuous
9	urgent	number of urgent packets	continuous
10	hot	number of “hot” indicators	continuous
11	num-failed-logins	number of failed login attempts	continuous
12	logged-in	1 if successfully logged in; 0 otherwise	symbolic
13	num-compromised	number of “compromised” conditions	continuous
14	root-shell	1 if root shell is obtained; 0 otherwise	continuous
15	su-attempted	1 if “su root” command attempted; 0 otherwise	contineous
16	num-root	number of “root” accesses	continuous
17	num-file-creation	number of file creation operations	continuous
18	num-shells	number of shell prompts	continuous
19	num-access-file	number of operations on access control files	continuous

Index	Feature Name	Description	Type
20	num-outbound-cmds	number of outbound commands in an ftp session	continuous
21	is-host-login	1 if the login belongs to the “hot” list; 0 otherwise	symbolic
22	is-guest-login	1 if the login is a “guest” login; 0 otherwise	symbolic
23	count	number of connections to the same host as the current connection in the past two seconds	continuous
24	srv-count	number of connections to the same service as the current connection in the past two seconds	continuous
25	serror-rate	percentage of connections that have “SYN” errors	continuous
26	srv-serror-rate	percentage of connections that have “SYN” errors	continuous
27	rerror-rate	percentage of connections that have “REJ” errors	continuous
28	srv-rerror-rate	percentage of connections to the same service	continuous
29	same-srv-rate	percentage of connections to the same service	continuous
30	diff-srv-rate	percentage of connections to different services	continuous
31	srv-diff-host-rate	percentage of connections to different hosts	continuous
32	dst-host-count	count for destination host	continuous
33	dst-host-srv-count	srv-count for destination host	continuous
34	dst-host-same-srv-rate	same-srv-rate for destination hos	continuous
35	dst-host-diff-srv-rate	diff-srv-rate for destination host	continuous
36	dst-host-same-src-port-rate	same-src-port-rate for destination host	continuous
37	dst-host-srv -diff-host-rate	diff-host-rate for destination host	continuous
38	dst-host-serror-rate	serror-rate for destination host	continuous
39	dst-host-srv-serror-rate	srv-serror-rate for destination host	continuous
40	dst-host-rerror-rate	rerror-rate for destination host	continuous
41	dst-host-srv-rerror-rate	nsrv-serror-rate for destination host	continuous

Appendix B - Classification Phase in RVM

In order to solve this equation, Maximum Likelihood (ML) can be applied however since the element of the vector W are mostly non-zero in RVM, due to the fact that RVM uses relevance vectors which are the vectors which have a certain distance to the hyperplane with a threshold, it shows the weakness of the RVM. Hence, we introduce a prior distribution for w and our best choice is to have it as a zero-mean Gaussian distribution. In other words we introduce a parameter α to which the probability of W is Dependant on. In Bayesian scheme, maximum likelihood estimation is adopted to calculate w. However, RVM is defined as the Gaussian prior probability distribution to avoid over fitting. In other words, a sparse weight prior distribution can be obtained in a way that different variance parameters are assigned for each weight.

$$p(w_i|\alpha_i) = \prod_{i=1}^{M} \mathcal{N}(w_i|0, \alpha_i^{-1}) = \prod_{i=0}^{N} \frac{\alpha_i}{\sqrt{2\pi}} \exp(-\frac{\alpha_i w_i^2}{2}) \tag{1}$$

where $\alpha = (\alpha_1, ..., \alpha_M)$ is a vector consisting of M hyper parameter. which are independent random variables.

$$2\alpha = [\alpha_0, \alpha_1, \alpha_2, \ldots, \alpha_N]^T \tag{2}$$

for which Gamma prior distribution is assigned

$$p(\alpha_i) = \prod_{i=0}^{N} Gamma(\alpha_i|a, b)$$

$$Gamma(\alpha_i|a, b) = \Gamma(a)^{-1} b^a \alpha^{a-1} \exp^{-b\alpha} \tag{3}$$

where $\Gamma(a) = t^{a-1}e^{-t}dt$ is the Gamma function and a and b are the constants and are set to zero. Therefore, based on the above equations the weights can be easily obtained. To better interpret it, RVM introduces the super parameter to each weight value leading to more sparse algorithm [72]. Therefore, the prediction formula in (4.25) will be changed to the following:

$$p(t_*|t) = \int p(t_*|w, \alpha, \sigma^2)p(w, \alpha, \sigma^2|t)dwd\alpha d\sigma^2 \tag{4}$$

The first term in the above equation can be rewritten as:

$$p(t_*|w, \alpha, \sigma^2) = p(t_*|w, \sigma^2) = \mathcal{N}(t_*|y(x_*; w), \sigma^2) \tag{5}$$

While the second term in the (4) can be calculated using:

$$p(w, \alpha, \sigma^2|t) = \frac{p(w, t, \alpha, \sigma^2)}{p(t)} = \frac{p(w, t, \alpha, \sigma^2)}{p(t, \alpha, \sigma^2)}\frac{p(t, \alpha, \sigma^2)}{p(t)} = p(w|t, \alpha, \sigma^2)p(\alpha, \sigma^2|t) \tag{6}$$

Similarly, in the equation (6) the first term and the second term can be found using equation (5.3) and (26)

$$p(w|t,\alpha,\sigma^2) = \frac{p(w,t,\alpha,\sigma^2)}{p(t,\alpha,\sigma^2)} = \frac{p(t|w,\alpha,\sigma^2)p(w,\alpha,\sigma^2)}{p(t|\alpha,\sigma^2)p(\alpha,\sigma^2)} =$$

$$\frac{p(t|w,\alpha,\sigma^2)p(w|\alpha,\sigma^2)p(\alpha,\sigma^2)}{p(t|\alpha,\sigma^2)p(\alpha,\sigma^2)} = \frac{p(w|\sigma^2)p(w|\alpha)}{p(t|\alpha,\sigma^2)} = \frac{p(t|w,\sigma^2)p(w|\alpha)}{\int p(t|w,\sigma^2)p(w|\alpha)dw} \tag{7}$$

$$p(t|w,\sigma^2) = \prod_{i=1}^{\mathcal{N}} \mathrm{N}(t_i|y(x_i;w),\sigma^2 = (2\pi\sigma^2)^{-\frac{N}{2}} \exp(-\frac{\|t-\Phi w\|^2}{2\sigma^2}) \tag{8}$$

$$p(w|\alpha) = \prod_{i=0}^{\mathrm{N}} \frac{\alpha_i}{\sqrt{2\pi}} \exp(-\frac{\alpha_i w_i^2}{2}) \tag{9}$$

$$p(w|t,\alpha,\sigma^2) = (2\pi)^{-\frac{\mathrm{N}+1}{2}} |\Sigma|^{-\frac{1}{2}} \exp\left\{-\frac{(w-\mu)^T\Sigma^{-1}(w-\mu)}{2}\right\} \tag{10}$$

$$p(t|\alpha,\sigma^2) = (2\pi)^{-\frac{\mathrm{N}}{2}} |\Omega|^{-\frac{1}{2}} \exp(-\frac{t^T\Omega^{-1}t}{2}) \tag{11}$$

in which Φ is a NxN or $Nx(N*M)$ matrices for single and multi kernel respectively and it is formed by $\Phi = [\phi_{x_1}, ..., \phi_{x_N}]^T$ with $\phi = [\phi_1(x_i - x_1), ..., \phi_M(x_i - x_N)]^T$

$$\Sigma = (\sigma^{-2}\Phi^T\Phi + A)^{-1} \tag{12}$$

$$\mu = \sigma^{-2}\Sigma\Phi^T t \tag{13}$$

$$A = \begin{bmatrix} \alpha_0 & 0 & 0 & \cdots & 0 \\ 0 & \alpha_1 & 0 & \cdots & 0 \\ 0 & 0 & \alpha_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \alpha_N \end{bmatrix} \tag{14}$$

$$\Omega = \sigma^2 I + \Phi A^{-1}\Phi^T \tag{15}$$

Hence, the probabilistic prediction formula will be:

$$p(t_*|t) = \int p(t_*|w, \alpha, \sigma^2)p(w|t, \alpha, \sigma^2)p(\alpha, \sigma^2|t)dwd\alpha d\sigma^2 \tag{16}$$

Here, we used delta function to approximate in order to solve $p(\alpha, \sigma^2|t)dwd\alpha d\sigma^2$ and since $p(\alpha, \sigma^2|t) = p(t|\alpha, \sigma^2)p(\alpha)p(\sigma^2)$

$$\alpha_{MP} = \arg\max_{\alpha}(p(t|\alpha, \sigma^2)p(\alpha) \tag{17}$$

$$\sigma^2_{MP} = \arg\max_{\sigma^2}(p(t|\alpha, \sigma^2)p(\sigma^2) \tag{18}$$

Since $p(\alpha)$ and $p(\sigma^2)$ have the fixed values, if we find the $(p(t|\alpha, \sigma^2)$, we can find $p(\alpha)$ and $p(\sigma^2)$. Hence:

$$p(t|\alpha, \sigma^2) = (2\pi)^{-\frac{N}{2}}|\Omega|^{-\frac{1}{2}}\exp(-\frac{t^T\Omega^{-1}t}{2}) \tag{19}$$

$$\begin{aligned} p(t_*|t) &= \int p(t_*|w, \alpha, \sigma^2)p(w|t, \alpha, \sigma^2)p(\alpha, \sigma^2|t)dwd\alpha d\sigma^2 \\ &\approx \int p(t_*|w, \alpha, \sigma^2)p(w|t, \alpha, \sigma^2)\delta(\alpha - \alpha_{MP})\delta(\sigma^2 - \sigma^2_{MP})dwd\alpha d\sigma^2 \\ &= \int p(t_*|w, \alpha_{MP}, \sigma^2_{MP})p(w|t, \alpha_{MP}, \sigma^2_{MP})dw \end{aligned} \tag{20}$$

The above equation is the product of the Gaussian functions, so we can get

$$p(t_*|t) = \mathrm{N}(t_*|y_*, \sigma_*^2) \tag{21}$$

$$y_* = \mu^T \phi(x_*) \tag{22}$$

$$\sigma_*^2 = \sigma_{MP}^2 + \phi(x_*)^T \sum \phi(x_*) \tag{23}$$

$$\phi(x_*) = [1, K(x_*, x_1), K(x_*, x_2), ..., K(x_*, x_N)]^T] \tag{24}$$

It is important to know that no delta function can be considered to effectively approximate the probability mass associated with this ridge, yet any point along it implies an identical predictive distribution and therefore it is reasonable to use delta function for approximation. Now the final step is to find the α_{MP} and σ_{MP}^2. using Maximum likelihood method, we have:

$$p(\alpha, \sigma^2|t) \propto p(t|\alpha, \sigma^2)p(\alpha)p(\sigma^2)(\alpha_{MP}, \sigma_{MP}^2) = \arg\max_{\alpha,\sigma^2} p(t|\alpha, \sigma^2) \tag{25}$$

$$p(t|\alpha, \sigma^2) = (2\pi)^{-\frac{N}{2}} |\Omega|^{-\frac{1}{2}} \exp(-\frac{t^T \Omega^{-1} t}{2}) \tag{26}$$

Since there is no closed-form solution for (25), in order to solve it, numerical solution can be adopted by taking the differentiation of (26) equals to zero. Then we have:

$$\alpha_i^{new} = \frac{\gamma_i}{\mu_i^2} \tag{27}$$

$$(\sigma^2)^{new} = \frac{||t - \Phi\mu||^2}{N - \sum_{i=0}^{N} \gamma_i} \tag{28}$$

$$\gamma_i = 1 - \alpha_i \Sigma_{i,i} \tag{29}$$

In this study we adopt two-class classification and we wish to predict the posterior probability of the classes for each instance in the test set. With utilizing the statistical convention and generalize the linear model by applying the logistic sigmoid link function $\sigma(y) = 1/(1 + e^{-y})$ to $y(X)$ and adopting Bernoulli distribution for $P(t|X)$, the likelihood can be written as:

$$p(t|w) = \prod_{n=1}^{N} \sigma\{y(X_n; W)^{t_n}[1 - \sigma\{y(X_n; W)\}]^{1-t_n} \tag{30}$$

Since for the classification problems the posterior probability of weights can not the calculated analytically and therefore no closed-form expression can be derived. The marginal likelihood $p(t|\alpha)$ can no longer be obtained by integrating the weights from (30) and hence, an iterative procedure needs to be used. One approach can be Laplace's approximation. Using the Laplace Theory [61], the maximum posterior probability estimation about w is equal to maximizing the following formula:

$$\log\{p(t|w)p(w|\alpha^2)\} = \sum\nolimits_{n=1}^{N} [t_n \log y_n + (1 - t_n)(1 - \log y_n)] - \frac{1}{2} W^T A W \tag{31}$$

Laplace's approach is basically a quadratic approximation to the log-posterior around its mode.

$$\nabla_w \nabla_w \log p(w|t, \alpha)|_{wMP} = -(\Phi^T B \Phi + A) \tag{32}$$

where B is equal to

$$B = \begin{bmatrix} \beta_0 & 0 & 0 & \cdots & 0 \\ 0 & \beta_1 & 0 & \cdots & 0 \\ 0 & 0 & \beta_2 & \dots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \dots & \beta_N \end{bmatrix} \tag{33}$$

$$\beta = \sigma(y(x_n))[1 - \sigma(y(x_n))] \tag{34}$$

$$\Sigma = (\Phi^T B \Phi + A)^{-1} \tag{35}$$

$$W_{MP} = \Sigma \Phi^T B t \tag{36}$$

The authors in [30] adopt another approximation. If α_i^* represent the maximum a posteriori (MAP) estimate of the hyper parameter α_i. The weights can be obtained by maximizing the following objective function:

$$J(w_1, w_2, ..., w_N) = \sum_{i=1}^{N} \log p(t_i|w_i) + \sum_{i=1}^{N} \log p(w_i|{\alpha_i}^*) \tag{37}$$

in which the likelihood of the class label and the prior on the parameters w_i is corresponded by first and second summation term. It is worth mentioning that only the samples associated with non-zero w_i called relevance vectors will contribute in the decision function. The gradient of the function J with respect to w is:

$$\nabla J = -A^* w - \Phi^T (f - t) \tag{38}$$

where $f = [\sigma(y(x_1))...\sigma(y(x_N))]^T$. The Hessian of J is

$$H = \nabla^2(J) = -\Phi^T B\Phi + A^* \tag{39}$$

$$\Sigma = -(H|_{W_{MAP}})^(- 1) \tag{40}$$

$$W_{MP} = \Sigma\Phi^T Bt \tag{41}$$

Appendix C - DoS Attack Types

Denial of Service Attack is any kind of attack that is then the availability and accessibility of the nodes to the network services. This attack prevents the legitimate nodes to use network assets. Denial of Service attacks are easier to cause damage in the network and therefore are very common on the Internet. In this Study, we focused on the Denial of Services attacks since they are very disruptive and harmful to the network. DoS Attacks in Mobile Ad hoc Networks can compromise any layer of the protocol stack. Denial of Service attacks affects the availability of the network and prevent legal access to the network services. When a malicious node denies network services by misdirecting, dropping or other attacks that threaten the availability of the network, Denial of Service attack has been launched on the network.

Neglect and Greed:

In this type of attack, the malicious node acknowledges the packets are received but refuse to forward it. It has its own way of prioritizing to decide which packets should be transmitted first. The severity of this attack highly depends on the location of the compromised node [45]. Therefore it can be derived that the number of ACK cannot be a reliable metric for IDS.

Selective Forwarding:

The selective forwarding attack manipulates flaws in the network layer to disrupt communication and choose which packets to forward. The selection of dropped packet can be random [45].

Misdirection:

There are two types of misdirection in wireless networks. In type 1, an adversary node receives the data packets and forward it to the wrong destination. It sends the traffic away from the intended destination node. However, in type 2, the adversary node denies the availability of

an existing route by sending a false RERR message and thus preventing access to the service while there is no alternative route [21, 60, 90].

Packet Loss:

During the data transmission between source and destination, there might be some intermediate nodes that relay the data packets. There might be some intermediate nodes that drop part or all of the forwarded packets instead of forwarding them to the next hop or the destination. Packet dropping attack is one of the very destructive denials of service attack in the network and very hard to detect with the low false positive rate. There a basically two types of packet drop or in general case, packet loss attack: malicious packet loss or legitimate packet loss which in this paper we are interested in detection of malicious packet loss. Packet Losses may occur due to three main reasons. First, the packet may get lost due to the existence and the presence of an attacker node. Second, selfish nodes that aim to preserve its energy refuse to collaborate in routing and drops the packets. Third, the packet loss can be due to other legitimate reasons that has been fully investigated in this study. An adversary node can randomly or selectively, known as a back hole and gray hole attack respectively drops either the control packets or data packets at the network layer which has a significant effect on the availability of the node [60] [15]. A malicious node can drop either the control packet or the data packet meaning that the malicious node is not relaying to packets [21]. Black hole and gray hole attacks are the very common kinds of packet dropping; however, with different dropping strategy. Black hole attack drops all the packets while gray hole attack drops packets selectively and it is more difficult to be mitigated [82].

Black hole In black hole attack, the attacker node compels transmitter node by advertising to have the shortest path to the destination. It advertises its availability without checking its

routing table through the RREP and it can easily intercept the packet and retain it. In a case, if this message reaches the transmitter prior to the actual reply message, the malicious node in the fake path, it will drop the packets and cause a denial of service to the network [44]. In other words, a malicious node can absorb all the network packets by claiming to have the shortest path to the destination. After receiving RREQ, it immediately responds with RREP to the source. Consequently, it convinced the source to have the best path to the destination and hence, forward all the packets to it. The adversary node drops all the packets after receiving them [83]. Black hole attack can sometimes be referred as sink-hole attack [8]. In this kind of attack, the malicious node introduce itself to have the shortest path to the destination and combine all nodes to forward their packets to it to relay them to the destination. However, while other nodes are forwarding their packets, malicious node drops the packets [33].

Grey hole In gray hole attack; however, the malicious node drops the packet only from selective nodes which make the detection even more difficult. In grey hole attacks, packets from certain users or packets to certain destinations will be dropped. Greyhole attack is one of the variations of the black hole attack and drops a selected bunch of packets. In this attack, the malicious node drops selected packets and forward the rest [4].

Wormhole In wormhole attack the presence of at least two malicious nodes is required that have better communication resources than regular nodes in the network [66]. The malicious nodes in this kind of attack establish a link among themselves and after receiving the packets, it forwards the packet to other malicious nodes instead of the destination. In other words, the malicious node tunnels the packets to other location in the network. Wormhole attack can cause disruption disorder and provide false routing information [75]. This attack diverts the packets selectively to another point in the network. It is one of the sophisticated attacks

in MANET networks. In this attack, the malicious node tunnels from one location in the network to another malicious node located several hops away from the source and send the captured packets to the other malicious node in the network [4]. The packets move faster in the tunnel between two malicious nodes than in reliable links. This attack is very difficult to be detected and relies on time delay and geographical location of the malicious nodes [4]. In other words, wormhole attacker records the packets from one location and tunnel them to another location in the network with the intention of applying some modification in the network [33]. A node should have high transmission power to be able to tunnel the data to another malicious node [33]. After the packets are received by another malicious node in the network, it will inject the data from that point to the network.

Sink-hole Sink-hole aims to capture all the traffic data through a malicious node using false information. Sink-hole attacks misroute the packet to the compromised node by providing the false information [33] wormhole attack, on the other hand, transmits the received packet to another point in the network and replay them into the network from that point. Sinking behavior happens when nodes do not cooperate in the routing and forwarding operations. In other words, when nodes fail to relay the packet nodes exhibit sinking behavior. The malicious node fails to forward the packets after sending RREP message to the initial route request from neighbor nodes. therefore all packets are sent to the malicious node which has the intention to forward them anywhere [49]. The attacker introduces itself to have the shortest path to the destination and therefore all nodes transmit their packets to this node while it drops all the received packets. The main intention of sink-hole attack is to attract all the traffic. Sink-hole attacks provide fake or modified routing information [75]. Sinking behavior is an active attack in which nodes do not cooperate in routing and drop either data packets or routing packets or

both [43]. In this attack, the malicious node first agrees to participate in routing and forward the packets; however, it refuses to forward the packets and fails to correctly cooperate in network forwarding operation.

Collision In order to enable transmission in wireless networks, the channel needs to be reserved for transmission through RTS/CTS packets and the legal node is able to transmit after RTS/CTS exchange. However, if an adversary node transmits in the range of the transmission, while the transmitter node is transmitting, a collision can occur in the wireless channel. The main purpose of the attacker node is to prevent access to a specific node or to exhaust the transmitter's resources by continuous transmission [60] [15]. A collision occurs when the intruder sends the wrong control message initially. In fact, in this type of attack, the intruder sends out a false RREP message indicating to have a route to the destination to all the neighboring nodes requesting it without checking its routing table. This way all neighboring nodes trust the malicious node and send their packets and malicious node intercept all the messages. In other words, if an adversary node starts transmitting while the channel is reserved for another node to transmit and it has already started transmitting collision occurs. The intention behind it is to prevent access to a specific node or to exhausting the transmitting node[1] [21]. Collisions can occur due to natural reasons or due to intentional disruption. When two nodes attempt to access the same shared medium at the same time, a natural collision occurs at the receiver node and the corrupted packets cannot be received by the receiver node. It is notable that natural collision highly depends on the traffic rate. However, there might be another node that does not follow the rules at MAC layer and cause an intentional collision. The intentional collision is caused

[1]Occurrence of collision due to either natural causes or due to the existence of an attacker does not change the fact that the node that does not receive the packets is malicious or disruptive because this node is the victim of collision attack

by MAC layer attack and occurs when the node does not follow the MAC layer rules [42]. The collision is very harmful in a way that even the collision of a single byte can damage the entire message. Collision attack not only discards packets and consumes energy, but it also causes corruption in control packets and data interference [75].

Malicious flooding:

`Hello` packets are used to inform nodes in the network about the new route. In this attack, once the node with high transmission power and good signal strength sends out the Hello messages to all the nodes in the network, the nodes suppose it is the neighboring node and start to transmit to it. This node will then drop the packet after receiving it. Malicious flooding deliberately exhausts the resources [75]. The ultimate goal of this attack is to exhaust the network resources such as bandwidth, computational power, and residual energy. This attack disrupts the network by sending out a large amount of fake routing control message which leads to severe congestion and power and space consumption [4].

Sybil Attack:

The attacker has multiple identities and it provides false and misleading messages and also causes resource exhaustion [75]. In this attack, the malicious node has multiple addresses and represent as a group of nodes. They can either create a new identity or steal the identity of other nodes. Therefore it prevents other nodes to use these identities. In this attack, the compromised node creates and presents multiple fake identities in an open-access medium which can either be stolen from other neighbors or be created by the adversary node [45]. The existence of the number of Sybil attackers can cause in negligence of a legitimate node in many network tasks which require collaboration like routing [8].

Spoofing:

Spoofing modifies the content of the routing packets and is the most difficult attack to be detected. Spoofing is done to deceive other nodes in the network by modifying the originator address [49].

Rushing:

In rushing the attacker rushes to send the RREQ message to take the path which contains the attacker node. To reduce the overhead due to route discovery, each node process only the received RREQ and deny any other duplicate packets that arrive later. Rushing takes advantage of this mechanism and quickly sends out the RREP [4].

Curriculum Vitae

Name:	Mahsa Bataghva Shahbaz
Post-Secondary Education and Degrees:	Semnan University Semnan, Iran Bachelor of Science in Engineering 2009 - 2013 University of Western Ontario London, ON Masters in Engineering Sceince 2015 - 2017
Honours and Awards:	Outstanding Presentation Award, NSERC CREATE, 2016 PSAC610 Academic Achievement Scholarship, 2017
Related Work Experience:	Teaching Assistant The University of Western Ontario 2015 - 2017

Publications:

1. A. Behnad, **M. Bataghva Shahbaz**, T. J. Willink and X. Wang, "Statistical Analysis and Minimization of Security Vulnerability Region in Amplify-and-Forward Cooperative Systems," in IEEE Transactions on Wireless Communications, vol. 16, no. 4, pp. 2534-2547, April 2017.

2. **M. B. Shahbaz**, Xianbin Wang, A. Behnad and J. Samarabandu, "On efficiency enhancement of the correlation-based feature selection for intrusion detection systems," 2016 IEEE 7th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON), Vancouver, BC, 2016, pp. 1-7.